

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Т. О. Терещенко, Ю.С. Ямненко

«СУЧАСНІ НАПРЯМКИ КОМП'ЮТЕРНОЇ ТА МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ» КОНСПЕКТ ЛЕКЦІЙ

**РОЗДІЛ 1. ОСНОВНІ ТЕНДЕНЦІЇ РОЗВИТКУ
КОМП'ЮТЕРНОЇ І МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ**

**РОЗДІЛ 2. ХАРАКТЕРИСТИКИ ARM І CORTEX
ПРОЦЕСОРІВ**

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для студентів,
які навчаються за спеціальністю 171 «Електроніка»,
спеціалізацією «Електронні компоненти і системи»*

Київ

КПІ ім. Ігоря Сікорського

2020

Рецензенти: *Михайлов С.Р.*, доцент кафедри електронних приладів та пристроїв, канд. техн. наук, доц.
 Відповідальний редактор: *Клен К.С.* доцент кафедри промислової електроніки, канд. техн. наук.

Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 6 від 31.01.2020 р.) за поданням Вченої ради факультету електроніки (протокол №01/2020 від 27.01.2020 р.)

Електронне мережне навчальне видання

Терещенко Тетяна Олександрівна, проф.
Ямненко Юлія Сергіївна, докт. техн. наук, проф.

«СУЧАСНІ НАПРЯМКИ КОМП'ЮТЕРНОЇ І МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ» КОНСПЕКТ ЛЕКЦІЙ

РОЗДІЛ 1. ОСНОВНІ ТЕНДЕНЦІЇ РОЗВИТКУ КОМП'ЮТЕРНОЇ І МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ РОЗДІЛ 2. ХАРАКТЕРИСТИКИ ARM І CORTEX ПРОЦЕСОРІВ

Сучасні напрямки комп'ютерної та мікропроцесорної техніки Розділ 1. Основні тенденції розвитку комп'ютерної і мікропроцесорної техніки. Розділ 2 Характеристики ARM і Cortex процесорів: конспект лекцій. [Електронний ресурс]: для студ. спеціальності 171 Електроніка, спеціалізації «Електронні компоненти та системи» /Т. О. Терещенко, Ю.С. Ямненко; КПІ ім. Ігоря Сікорського; уклад,– Електронні текстові данні 1 файл: 5,248 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2020. – 68 с.

Навчальний посібник «Сучасні напрямки комп'ютерної та мікропроцесорної техніки. Розділи 1 та 2: конспект лекцій ставить на меті надання студентам ґрунтовних знань відносно основних тенденцій розвитку комп'ютерної і мікропроцесорної техніки та набуття студентом здатності використовувати основні знання по архітектурі сучасних мікроконтролерів та створення нових пристроїв мікропроцесорної техніки В конспекті наведено матеріали лекцій перших 2двох розділів згідно навчального плану,

© Т. О. Терещенко, Ю.С. Ямненко, 2020

© КПІ ім. Ігоря Сікорського, 2020

Вступ.....	4
РОЗДІЛ 1 Основні тенденції розвитку комп'ютерної і мікропроцесорної техніки	5
Тема 1.1. Історія розвитку комп'ютерної і обчислювальної техніки.....	5
Тема 1.2. Історія розвитку мікропроцесорної техніки	10
Тема 1.3 Покоління комп'ютерів	14
Тема 1.4 Суперкомп'ютери	17
Тема 1.5 Нейрообчислювачі	20
Тема 1.6 Квантові комп'ютери	25
Тема 1.7 Оптичні комп'ютери	28
Тема 1.8 Біокомп'ютери на основі ДНК	30
РОЗДІЛ 2 Характеристики ARM і Cortex процесорів.....	32
Тема 2.1 Передумови створення ARM. Процесори Cortex.....	32
Тема 2.2. Шини ARM-процесорів Організація пам'яті у мікроконтролерах....	45
Тема 2.3. Порти введення/ виведення. Контролери NVIC та EXTI. Таймери ..	53
Перелік посилань.....	66

Вступ

Метою кредитного модуля є здатність використовувати основні знання принципів роботи сучасних мікропроцесорів і комп'ютерів, архітектурно-структурні ідеї, які покладено в основу сучасних засобів обробки інформації, проведення порівняльного аналізу різних архітектур і мікропроцесорів (мікроконтролерів), методів та засобів розробки апаратної частини та програмного забезпечення комп'ютерами та набуття практичних навичок по проектуванню систем з мікропроцесорами та комп'ютерами.

Основні завдання кредитного модуля.

Згідно з вимогами програми навчальної дисципліни студенти після засвоєння кредитного модуля мають продемонструвати такі результати навчання:

знання:

- основних тенденцій розвитку комп'ютерної та мікропроцесорної техніки;
- загальних принципів побудови та функціонування новітніх комп'ютерних та мікроконтролерних систем;
- методів та засобів розробки апаратної частини та програмного забезпечення систем з мікропроцесорами та комп'ютерами

вміння:

самостійно працювати з науково-технічною літературою по мікропроцесорним та комп'ютерним системам різноманітного призначення;

- провести порівняльний аналіз різних архітектур і мікропроцесорів (мікроконтролерів),
- мати практичні навички по вибору елементної бази мікроконтролерів і керуючих платформ
- використовувати набуті знання при проектуванні і конструюванні новітніх комп'ютерних та мікропроцесорних систем
- розробляти програмне забезпечення комп'ютерних та мікропроцесорних систем.

РОЗДІЛ 1 Основні тенденції розвитку комп'ютерної і мікропроцесорної техніки

Тема 1.1. Історія розвитку комп'ютерної і обчислювальної техніки

Причини виникнення обчислювальних пристроїв відображено на рис.1.1. Необхідність здійснювати торгові операції, проводити землемірні роботи, управляти запасами врожаю, стежити за астрономічними циклами призвело до прагнення полегшити обчислення та розробки різних інструментів, від рахункових паличок і абаків до калькуляторів та комп'ютерів.

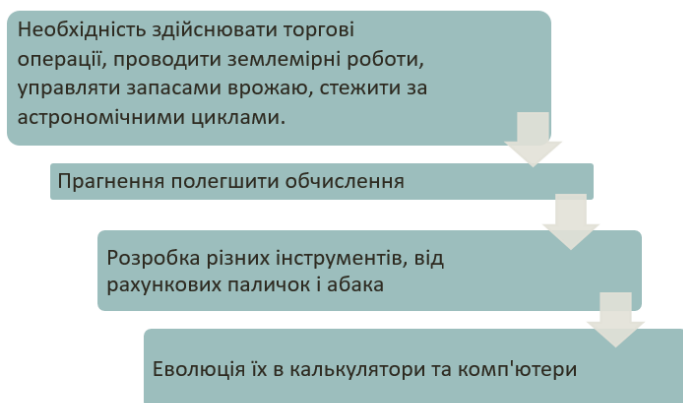


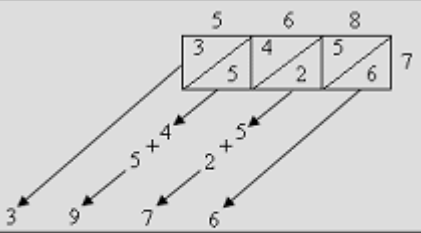
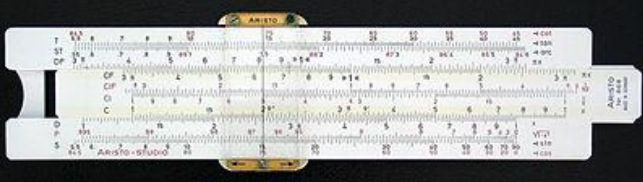


Рис.1.1 Причини виникнення обчислювальних пристроїв

Основні етапи всесвітньої історії комп'ютеробудування

Кожен новий етап створення обчислювальних пристроїв базувався на наукових досягненнях свого часу. Основні етапи всесвітньої історії комп'ютеробудування відображена в табл.1.1

Таблиця 1.1-Основні етапи історії комп'ютеробудування

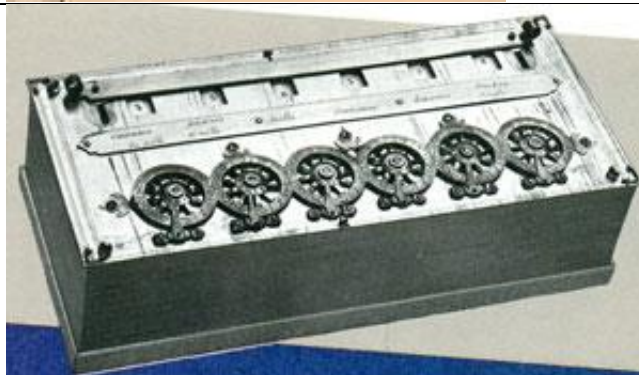
Наукові досягнення	Пристрої
<p>400 років до н. е. Аристотель заклав основи математичної логіки, ввів поняття змінної в логіці, застосував букви для позначення понять.</p> <p>Близька 820 р Математик аль-Хорезмі ввів поняття алгоритму (так званого «аль-хоризмой») і десяткову систему числення</p>	<p>3000 років до н.е.</p> <p>Абак (Abacus) - найстаріше з відомих рахункових пристроїв, використовувалося в Давньому Римі, Греції та Китаї, «прадід» сучасних дерев'яних рахунків. В Стародавньому Римі називався calculi або abakuli.</p> 
<p>IX-X ст. У Європі поширюються арабські цифри, в яких є поняття нуля і позиційність</p> <p>1612 рік. Джон Нейпер вводить «десятова кома», розробляє таблиці логарифмів, винаходить лічильні кістки. На їх основі потім розроблено номограми, що дозволяють простими геометричними операціями вирішувати рівняння</p>	 <p>Палочки Непера</p>  <p>Умножение решеткой 568 x 7 = 3 976</p>
<p>1623 рік Незабаром після введення в обіг логарифмів у 1623 р. з'явилася логарифмічна лінійка, яку винайшов англійський математик Едмонд Гантер</p>	 <p>Типова логарифмічна лінійка. Показано обчислення добутку $1,3 \times 2 = 2,6$</p>

1624 рік

Базуючись на роботах Нейпера, професор Вільгельм Шиккард побудував першу машину, що виконувала чотири арифметичні дії.

**1642 рік** Блез Паскаль створює механічну машину, яка могла складати числа.

Паскаль адресував свій винахід батька - податковому збирачеві. Числа вводилися в машину за допомогою набірних коліщаток, які відповідали десятковим розрядам числа. При введенні числа коліщатка прокручувалися до відповідної цифри. Зробивши повний оборот, надлишок над цифрою 9 коліщатко переносило на сусідній розряд, зрушуючи сусіднє колесо на 1 позицію

**1673 рік**

Готфрід фон Лейбніц побудував механічну лічильну машину, яка, замість коліщат використовувалися циліндри з нанесеними на них цифрами. Кожен циліндр мав дев'ять ряд ОРТдів виступів або зубців. При цьому перший ряд містив один виступ, другий ряд — два виступи і так аж до дев'ятого ряду, що містив відповідно дев'ять виступів. Циліндри з виступами були рухомими



1833 р. Чарльз Беббідж створив автоматичну машину, що працювала від пари. З її допомогою можна було легко складати, віднімати, множити і ділити шестизначні числа. Далі винахідник розвиває ідею нової аналітичної машини (Analytical Engine). Вона включала центральний процесорний пристрій, пам'ять і перфокарти, на яких набивали певні програми. Пристрій змогло б оперувати з двадцятизначними числами

Демонстраційна модель обчислювальної частини різницевої машини Чарльза Беббіджа

лорда Байрона - леді Ада Байрон Кінг (1815-1852) описала роботу машини Беббіджа і створила для неї першу програму. Леді Кінг прийнято вважати першим програмістом.

Перша програма – обчислення чисел Бернуллі включала цикли та умовні переходи) Введення даних і програми здійснювалось з перфокарт

Мова програмування Ада (1979) в її честь



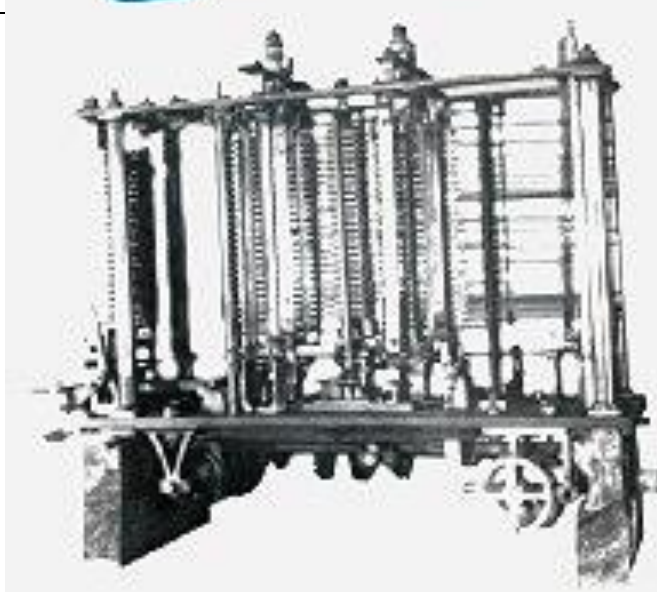
1880 рік

В. Т. Однер створює в Росії арифмометр з зубчаткою зі змінним числом зубців, а в 1890 р налагоджує масовий випуск удосконалених арифмометрів, які в першій чверті XX ст. були основними математичними машинами, що знайшли застосування в усьому світі. Їх модифікація «Фелікс» випускалася в СРСР до 70-х років



1896 рік

Машина для перепису населення інженера Германа Холлерита була одним з перших електронних обчислювальних пристроїв, які використовували перфокарти. Ідеї, закладені в ній, знайшли застосування і в сучасних обчислювальних пристроях. Окрилений успіхом, Холлерит відкрив власну компанію, з неї пізніше утворилася IBM.



Зазначимо, що причинами створення різницевої машини Чарльза Беббіджа стали помилки в таблицях логарифмів, обумовлені людським фактором. Гаспар де Проні, керівнику бюро перепису при французькому уряді з 1790 по 1800 рік, якому було доручено вивірити і поліпшити логарифмічні тригонометричні таблиці для підготовки до введення метричної системи, запропонував розподілити роботу за трьома рівнями. На верхньому рівні група

великих математиків займалася виведенням математичних виразів, придатних для чисельних розрахунків. Друга група обчислювала значення функцій для аргументів, віддалених один від одного на п'ять або десять інтервалів. Підраховані значення входили в таблицю в якості опорних. Після цього формули відправляли третьої, найбільш численної групи, члени якої проводили рутинні розрахунки і іменувалися «обчислювачами». Від них вимагалось лише акуратно додавати і віднімати в послідовності, визначеній формулами, отриманими від другої групи. Беббідж запропонував створити машину, здатну замінити третю групу - обчислювачів. Як математику, Беббіджу був відомий метод апроксимації функцій многочленами і обчисленням кінцевих різниць. З метою автоматизації цього процесу він почав проектувати машину, яка так і називалася - різницева. Ця машина повинна була вміти обчислювати значення многочленів до шостого ступеня з точністю до 18-го знаку.

Серед інших винаходів, що вплинули на розвиток комп'ютеробудування варто відмітити:

- Передача повідомлень по дротах. Самуель Морзе передав по повідомлення з Вашингтона в Балтімор, що знаходиться на відстані 58 кілометрів (1844 рік)
- Механічний пристрій для автоматизації Петербурзьким учителем музики Куммером запропоновано механічний пристрій для автоматизації обчислень, що серійно випускалися (з різними модифікаціями) аж до 70-х років XX ст. (1846 рік)
- Телефон. (Олександр Грехем Белл, 1876 рік)
- Електронно-променева трубка (Дж. Томсон, 1897)
- Вакумні лампи – діод, тріод (1906)
- Тригер – пристрій для зберігання біта (М. А. Бонч-Бруєвич, 1918).
- калькулятор, який оперує з двійковими числами. (Інженери Абрахам і Блох, 1918 рік)
- аналоговий комп'ютер. (Массачусетський технологічний інститут, 1919 рік)
- Використання математичної логіки в комп'ютерах (К. Шеннон, 1936)
- перша обчислювальна машина на основі двійкової системи числення (Джордж Стібітц, лабораторія Bell Telephone, 1937)
- Повністю механічна програмована цифрова машину Z1. Саме цей пристрій сьогодні називають першим у світі комп'ютером (Німецький інженер Конрад Цузе, 1938 рік)
- Перший тоновий генератор (Білл Хьюлетт і Девід Паккард, 1939 рік)
- Розробка теорії алгоритмів, математичну модель комп'ютера, відому як «Машина Тюрінга», (Англійський математик Алан Тюрінг, 1939).
- перший електронний програмований калькулятор Z3 (Цузе , 1941 рік)

- перший в США електронний цифровий комп'ютер ABC (Atanasoff-Berry Computer). (Викладачі університеті штату Айова Джон Атанасов і Кліффорд Беррі 1942 рік)
- Релейно-механічний комп'ютер Mark I (Говард Айкен, 1944). Комп'ютер оперував з 23-значними числами. Розробник Говард Айкен використовував його для обчислень під час роботи над створенням атомної бомби. Перший комп'ютер в США: – довжина 17 м, вага 5 тон – 75 000 електронних ламп – 3000 механічних реле – додавання – 3 секунди, ділення – 12 секунд 1944 рік

Тема 1.2. Історія розвитку мікропроцесорної техніки

Перші МП Intel

Перший мікропроцесор i4004 був виготовлений в 1971р і з тих пір фірма Intel міцно утримує лідируюче положення на сегменті ринку. Найбільш успішний був проект розробки i8080. Саме на ній був заснований комп'ютер "Альтаір", для якого Б. Гейтс написав свій перший інтерпретатор Basic. Класична архітектура i8080 справила величезний вплив на подальший розвиток однокристальних мікропроцесорів.

Справжнім промисловим стандартом для PC став мікропроцесор i8088, який був анонсований Intel в червні 1979 р. У 1981 р фірма IBM вибрала цей процесор для свого PC. Спочатку мікропроцесор i8088 працював на частоті 4.77 МГц і мав швидкодію близько 0.33 MIPS, однак потім були розроблені його клони, розраховані на більш високу тактову частоту 8 МГц.

Мікропроцесор i8086 з'явився в липні 1978 р, став популярний завдяки комп'ютеру Compaq Dec Pro, рис. 1.2

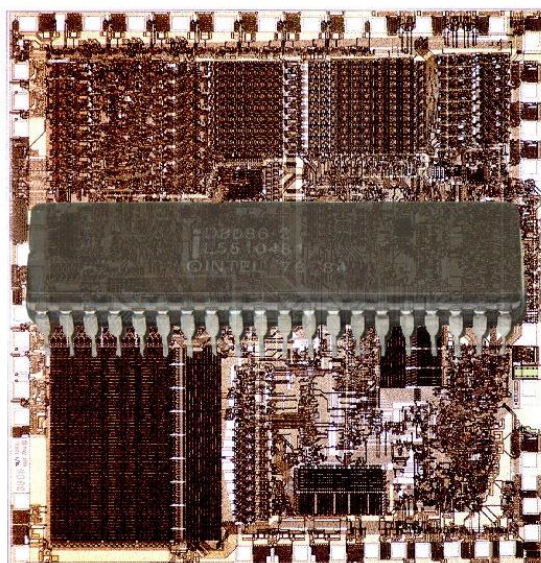


Рис.1.2.МП 8086: перший процесор для ПК

Цей 16-бітний процесор міг працювати з 1 Мбайт пам'яті по зовнішній 20-бітній адресній шині. Тактова частота 10 МГц.

Перші ПК використовували похідну процесора 8088, що яка мала всього 8-бітну зовнішню шину даних. Що цікаво, системи управління в американських шатлах використовують процесори 8086, і NASA довелося в 2002 році купувати процесори через eBay, оскільки Intel їх більше не виробляла.

Технічні характеристики процесора 8086 наведено в табл.1.2

Таблиця 1.2 - Технічні характеристики процесора 8086

Intel 8086	
Дата випуску	1979
Архітектура	16 біт
Шина даних	16 біт
Шина адреси	20 біт
Макс. об'єм пам'яті	1 Мбайт
Кеш L1	-
Кеш L2	-
Тактова частота	4,77-10 МГц
FSB	Така сама як і в CPU
FPU	8087
SIMD	
Техпроцесс	3 000 нм
Кількість транзисторів	29 000
Напруга живлення	5 В
Площа кристалу	16 мм ²
Сокет	40-контактний

32-бітний МП з кеш-пам'яттю i386

Intel 80836 став першим процесором x86 з 32-бітної архітектурою, рис.3. Вийшло кілька версій цього процесора. Дві найбільш відомі: 386 SX (Single-word eXternal), який використовував 16-бітну шину даних, і 386 DX (Double-word eXternal) з 32-бітної шиною даних. Можна відзначити ще дві версії: SL, перший процесор

x86 з підтримкою кеша (зовнішнього) і 386EX, який використовувався в космічній програмі (наприклад, телескоп "Хаббл" використовує цей процесор).

Зовнішній вигляд МП представлено на рис. 1.3, а технічні характеристики наведено в табл.1.3.

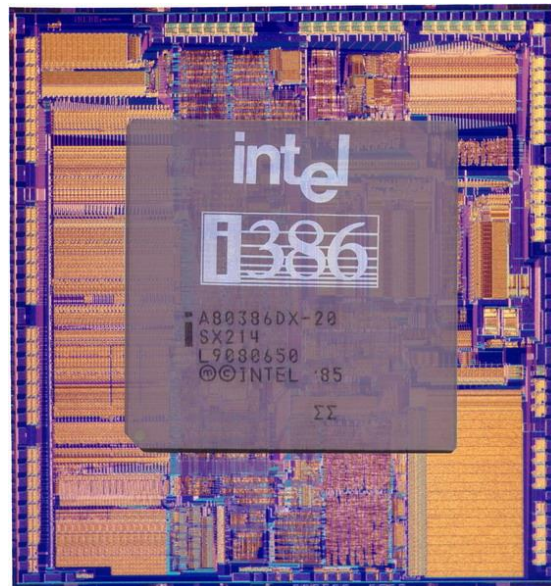


Рис. 1.3 МП 80386

Таблиця 1.3 - Технічні характеристики процесора 80386

Intel 80386	
Дата випуску	1985
Архітектура	32 біт
Шина даних	32біт
Шина адреси	32 біт
Макс. об'єм пам'яті	4096 Мбайт
Кеш L1	зовнішній
Кеш L2	-
Тактова частота	16-33 МГц
FSB	Така сама як і в CPU
FPU	80387

SIMD	-
Техпроцесс	1500-1000 нм
Кількість транзисторів	275 000
Енергоспоживання	2 Вт @ 33 МГц
Напруга живлення	5 В
Площа кристалу	42 мм ²
Сокет	132-контактний
Кодова назва	P3

Архітектура 32 і 64 розрядних МП

Типи архітектур МП наведено в табл.1.4

Таблиця 1.4 – Архітектури 32 і 64 розрядних МП

Назва архітектури	Процесори
x86/IA32 (32 біт)	80386 • 80486 • Pentium • Pentium Pro • Pentium II • Pentium III • Pentium 4 • Pentium M • Centrino • Core • Celeron M • Celeron D •
x86-64/EM64T (64 біт)	Pentium 4 (Some) • Pentium D • Pentium Extreme Edition •
Інші	Original Itanium — iAPX 432 — RISC: i860 • i960 • XScale

Процесори Pentium Перший у серії Pentium був процесором Intel п'ятого покоління і прийшов на зміну Intel 80486 У процесорах Pentium реалізується архітектура IA-32 (Intel Architecture-32), загальна для всіх 32-розрядних мікропроцесорів Intel:

- Pentium Pro — перший процесор шостого покоління, або ядро P6
- Pentium II — комерційно успішний процесор шостого покоління
- Pentium III — варіант Pentium II з дещо розширеною системою команд
- Pentium 4 — процесор сьомого покоління
- Pentium M — мобільний варіант чипа, на базі вдосконаленого ядра P6
- Pentium D — двоядерний процесор Pentium 4

Тема 1.3 Покоління комп'ютерів

Розрізняють наступні покоління комп'ютерів:

- I. 1945 – 1955 електронно-вакумні лампи
- II. 1955 – 1965 транзистори
- III. 1965 – 1980 інтегральні мікросхеми
- IV. з 1980 по теперішній час - великі і надвеликі інтегральні схеми (BIC і NBIC)
- V. На даний момент термін «п'яте покоління» є невизначеним і застосовується в багатьох сенсах, наприклад, при описі систем хмарних обчислень.

ЕОМ першого покоління

Характерними рисами ЕОМ першого покоління є застосування електронних ламп у цифрових схемах, великі габарити, а також трудомісткий процес програмування. Насправді, ЕОМ першого покоління розміщувалися у великих машинних залах, споживали багато електроенергії та вимагали охолодження за допомогою потужних вентиляторів. Програми для цих ЕОМ потрібно було складати у машинних кодах, і це могли робити тільки фахівці, що знали детально пристрій ЕОМ

Прикладом ЕОМ першого покоління є комп'ютер ЕНІАК, рис.1.4

Перший комп'ютер загального призначення на електронних лампах: ЕНІАК (1946) Electronic Numerical Integrator And Computer Дж. • довжина 26 м, маса 35 тон • додавання – $1/5000$ с, ділення – $1/300$ с • десяткова система числення • 10 –розрядні числа

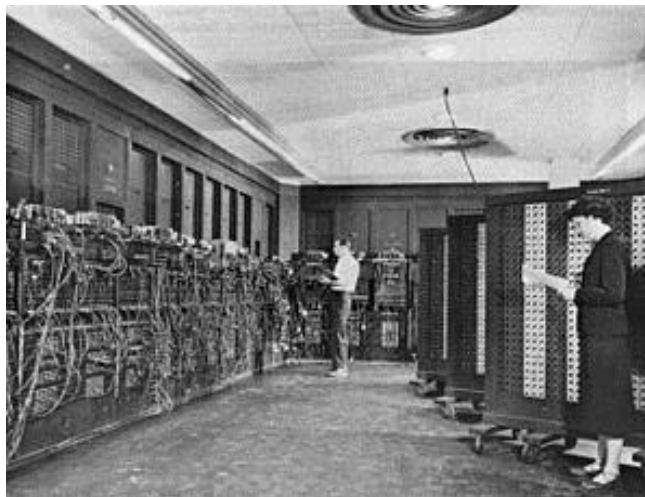


Рис. 1.4. Перший комп'ютер загального призначення на електронних лампах

ЕОМ другого покоління

Коли в середині 50 х років на зміну електронним лампам прийшли напівпровідникові прилади, почалося переведення ЕОМ на напівпровідники. Напівпровідникові прилади були, по перше, значно компактнішими, ніж їхні лампові попередники. По друге, вони мали триваліший термін служби. По третє, споживання енергії в ЕОМ на напівпровідниках було істотно нижчим. З

упровадженням цифрових елементів на напівпровідникових приладах почалося створення ЕОМ другого покоління. ЕОМ другого покоління відрізняються також застосуванням алгоритмічних мов програмування.

Завдяки застосуванню більш досконалої елементної бази почали створюватися невеликі ЕОМ, сталося розподілення обчислювальних машин на великі, середні й малі.

В Україні першою малою ЕОМ стала машина «Днепр 1», серійне виробництво якої було налагоджено на заводі «Арсенал» (м. Київ). ЕОМ «Днепр 1» передувала унікальній за своєю архітектурою машині «Мир 1», розробленій в 1965 р. в Інституті кібернетики (керівник В. М. Глушков). Машина «Мир 1» та її наступна модифікація «Мир 2» передбачались для інженерних розрахунків, які виконував на ЕОМ сам користувач без допомоги оператора

Вигляд найкращою радянської ЕОМ другого покоління БЕСМ – 6 наведений на рис. 1.5



Рис.1.5. БЕСМ – 6

ЕОМ другого покоління БЕСМ – 6 мала наступні характеристики:

- Тактова частота: 9 МГц
- Елементна база: біполярні транзистори П416, напівпровідникові діоди
- Кількість транзисторів: 60000, кількість діодів: 180000
- Розрядність даних: 48 біт
- Розрядність адрес: 15 біт
- Швидкодія: приблизно 1 MIPS
- Центральний процесор: конвеєрної архітектури
- Оперативна пам'ять: 8 банків (з одночасним доступом), на феритових кільцях
- Споживана потужність: близько 50 кВт

ЕОМ третього покоління

Характерними рисами ЕОМ третього покоління є застосування інтегральних схем і можливість використання розвинутих мов програмування (мов високого рівня)

- швидкодія до 1 млн. операцій в секунду
- оперативна пам'ять – сотні Кбайт
- операційні системи – управління пам'яттю, пристроями, часом процесора
- мова програмування Бейсик (1965), Паскаль (1970, Н. Вірт), Сі (1972, Д. Рітчі)
- сумістність програм

У третьому поколінні з'явилися великі серії ЕОМ, що розрізняються за своєю продуктивністю і призначенням. Це родина великих і середніх машин IBM 360/370, розроблених у США. У Радянському Союзі й у країнах РЕВ були створені аналогічні серії машин: ЄС ЕОМ (Єдина Система ЕОМ, машини великі і середні), СМ ЕОМ (Система Малих ЕОМ) і «Електроніка» (система мікро ЕОМ).

Комп'ютери четвертого покоління

Комп'ютери четвертого покоління- це на великих і надвеликих інтегральних схемах (БІС, НБІС)

У процесі вдосконалення мікросхем збільшувалася їхня надійність і щільність розміщених в них елементів. З'явилися великі інтегральні схеми (БІС), у яких на один квадратний сантиметр припадає декілька десятків тисяч елементів. На основі БІС були розроблені ЕОМ наступного четвертого покоління

ЕОМ четвертого покоління характеризуються застосуванням мікропроцесорів, побудованих на великих інтегральних схемах. Вплив персональних комп'ютерів на уявлення людей про обчислювальну техніку виявився настільки великим, що поступово з ужитку зник термін «ЕОМ», а його місце зайняло слово «комп'ютер»

Характерні особливості та характеристики комп'ютерів четвертого покоління наступні:

- суперкомп'ютери
- персональні комп'ютери
- поява користувачів-непрофесіоналів,
- необхідність «дружнього» інтерфейса
- більше 1 млрд. операцій в секунду
- оперативна пам'ять – до декількох гігабайт
- багатопроцесорні системи
- комп'ютерні мережі
- мультимедіа (графіка, анімація, звук))

П'яте покоління

Починаючи із середини 90-х років, у потужних комп'ютерах застосовуються супермасштабні ВІС, які вміщують сотні тисяч елементів на квадратний сантиметр. Багато фахівців почали говорити про комп'ютери п'ятого покоління. Характерною рисою комп'ютерів п'ятого покоління повинно бути використання штучного інтелекту і природних мов спілкування. Передбачається, що обчислювальні машини п'ятого покоління будуть легко керуваними. Користувач зможе голосом подавати команди машині

У покоління (проект 1980-х, Японія)

Мета – створення суперкомп'ютера з функціями штучного інтелекту обробка знань за допомогою логічних засобів (мова Пролог), причому передбачалося розробити

- надвеликі бази даних
- використання паралельних обчислень
- розподілені обчислення
- голосові повідомлення з комп'ютером
- поступова заміна програмних засобів на апаратні

Однак проект зіткнувся з наступними проблемами:

- ідея саморозвитку системи провалилась
- невірна оцінка балансу програмних і апаратних засобів
- традиційні комп'ютери досягнули більшого
- ненадійність технологій
- витрачено 50 млрд. йєн

Проблеми і перспективи створення комп'ютерів 5 покоління

Проблеми:

- наближення до фізичної межі швидкодії
- складність програмного забезпечення приводить до зниження надійності

Перспективи:

- квантові комп'ютери □ ефекти квантової механіки □ паралельність обчислень □ 2006 – комп'ютер з 7 кубіт
- оптичні комп'ютери («заморожене світло»)
- біокомп'ютери на основі ДНК □ хімічна реакція з участю ферментів □ 330 трлн. операцій в секунду

Тема 1.4 Суперкомп'ютери

Суперкомп'ютер (supercomputer) — загальний термін, який використовується для позначення класу існуючих найпотужніших комп'ютерних систем.

Суперкомп'ютери, зазвичай, використовуються при вирішенні складних наукових та інженерних задач, які вимагають виконання великої кількості математичних операцій та(чи) працюють з великими об'ємами даних.

З систем, що працюють в Україні, слід відзначити наступні:

- Центр суперкомп'ютерних обчислень НТУУ «КПІ»
- Суперкомп'ютерний обчислювальний центр (СОЦ) на базі високопродуктивних кластерних систем СКІТ Інституту кібернетики ім. В. М. Глушкова

Вигляд одного з перших суперкомп'ютерів ILLIAC-IV (США) наведено на рис. 1.6.

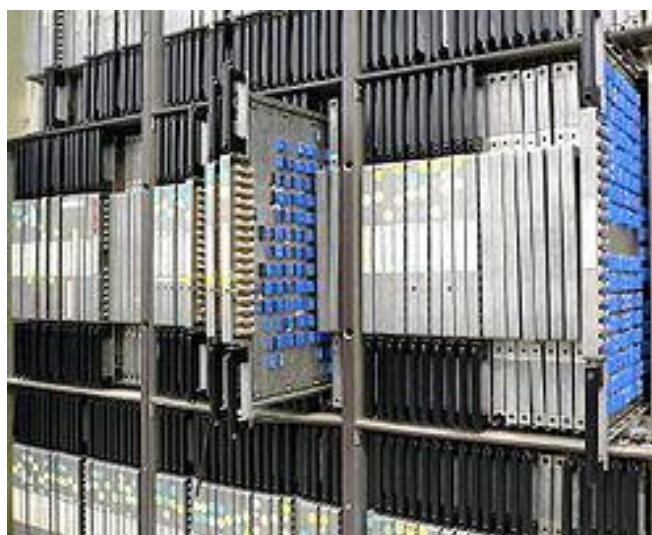


Рис. 1.6 Вигляд суперкомп'ютера ILLIAC-IV

Основні етапи створення та характеристики суперкомп'ютерів

Основні етапи створення та характеристики перших суперкомп'ютерів наступні:

- 1972. ILLIAC-IV (США) • 20 млн. оп/с • багатопроцесорна система
- 1976. Cray-1 (США) • 166 млн. оп/с • пам'ять 8 Мб • векторні обчислення
- 1980. Ельбрус-1 (СРСР) • 15 млн. оп/с • пам'ять 64 Мб
- 1985. Ельбрус-2 • 8 процесорів 125 млн. оп/с • пам'ять 144 Мб • водяне охолодження
- 1985. Cray-2 2 млрд. оп/с
- 1989. Cray-3 5 млрд. оп/с
- 1995. GRAPE-4 (Японія) 1692 процесора 1, 08 трлн. оп/с
- 2002. Earth Simulator (NEC) 5120 процесорів 36 трлн. оп/с
- 2007. Blue. Gene/L (IBM) 212 992 процесора 596 трлн. оп/с

Галузі застосування суперкомп'ютерів

Суперкомп'ютери використовуються в усіх сферах, де потрібен величезний обсяг складних обчислень:

- Математичні проблеми:
 - Криптографія
 - Статистика
- Фізика високих енергій:
 - процеси всередині атомного ядра, фізика плазми, аналіз даних експериментів, проведених на прискорювачах
 - розробка і вдосконалення атомної і термоядерної зброї, управління ядерним арсеналом, моделювання ядерних випробувань
 - моделювання життєвого циклу ядерних паливних елементів, проекти ядерних і термоядерних реакторів
- Наука про Землю:
 - прогноз погоди, стану морів і океанів
 - пророкування кліматичних змін та їх наслідків
 - дослідження процесів, що відбуваються в земній корі, для передбачення землетрусів і вивержень вулканів
 - аналіз даних геологічної розвідки для пошуку і оцінки нафтових і газових родовищ, моделювання процесу вироблення родовищ
 - моделювання розтікання річок під час паводку, розтікання нафти під час аварій
- Обчислювальна біологія:
 - фолдінг білка,
 - розшифровка ДНК
- Обчислювальна хімія та медицина: пошук і створення нових ліків
- Фізика:
 - газодинаміка: турбіни електростанцій, горіння палива, аеродинамічні процеси для створення досконалих форм крила, фюзеляжів літаків, ракет, кузовів автомобілів
 - гідродинаміка: потік рідин по трубах, по руслах річок
 - матеріалознавство: створення нових матеріалів із заданими властивостями, аналіз розподілу динамічних навантажень в конструкціях, моделювання креш-тестів при конструюванні автомобілів

Розвиток суперкомп'ютерної техніки

Компанія IBM наразі працює над створенням суперкомп'ютера Blue Waters, здатного досягти продуктивності 16 петафлопс. (16x10¹⁵ Floating point Operations Per Second)

В 2012 р. IBM випустили суперкомп'ютер IBM Sequoia з потужністю 20 петафлопс. Цей комп'ютер використовується для моделювання випробувань ядерної зброї.

Уже здійснено запуск суперкомп'ютерного кластера на базі IBM BladeCenter з потужністю 10 терафлопс в Казахстансько-Британському технічному університеті.

В 2019 р. очікується поява суперкомп'ютера з продуктивністю, що буде вимірюватись уже в ексафлопсах (10^{18} операцій за секунду) (Білорусь.)

Тема 1.5 Нейрообчислювачі

Передумови виникнення нейрокомп'ютерів

Є ряд задач, наприклад, проблеми розпізнавання образів, виконання прогнозів, оптимізації, асоціативній пам'яті і керування, які важко вирішуються традиційними обчислювачами на відміну від людського мозку. Нейронні мережі виникли з досліджень в області штучного інтелекту, а саме, зі спроб відтворити здатність біологічних нервових систем навчатися і виправляти помилки. Штучні нейромережі є електронними моделями нейронної структури мозку, який головним чином навчається з досвіду.

Штучні нейронні системи ґрунтувалися на високорівневому моделюванні процесу мислення на звичайних комп'ютерах. Скоро стало ясно, щоб створити штучний інтелект, необхідно побудувати систему з схожою на природну архітектурою, т. е. перейти від програмної реалізації процесу мислення до апаратної.

Нейрокомп'ютер

Нейрокомп'ютер — пристрій переробки інформації на основі принципів роботи природних нейронних систем Ці принципи були формалізовані, що дозволило говорити про теорії штучних нейронних мереж.

Проблематика нейрокомп'ютерів полягає в побудові реальних фізичних пристроїв, що дозволить не просто моделювати штучні нейронні мережі на звичайному комп'ютері, але так змінити принципи роботи комп'ютера, що стане можливим говорити про те, що вони працюють відповідно до теорії штучних нейронних мереж.

Інтерес до реалізації нейроподібних систем, здатних вирішувати інтелектуальні завдання, зріс в кінці п'ятдесятих років.

В 1957 р. нейрофізіолог Френк Розенблат запропонував схему пристрою, що моделює процес людського сприйняття, і назвав його «перцептроном». Перцептрон передавав сигнали від фотоелементів, що являють собою сенсорне поле, в блоки електромеханічних елементів В 1960 року в Корнельському університеті було продемонстровано перший нейрокомп'ютер— «Марк-1», що був здатен розпізнавати деякі з літер англійського алфавіту.

Логічна схема елементарного перцептрону наведена рис.1.7.

Елементарний перцептрон складається з елементів трьох типів: S-елементів, A-елементів та одного R-елементу. S-елементи — це шар сенсорів, або рецепторів. Ваги зв'язків S-A можуть мати значення -1 , 1 або 0 (тобто відсутність зв'язку). Ваги зв'язків A-R W_i можуть мати будь-яке значення

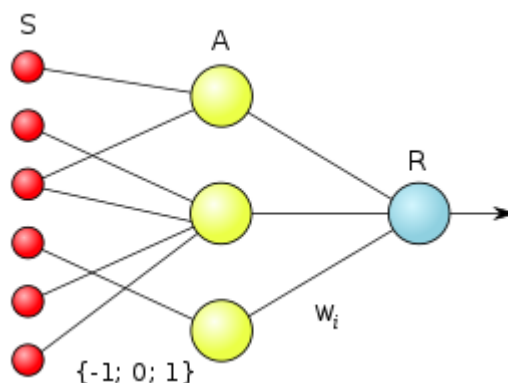
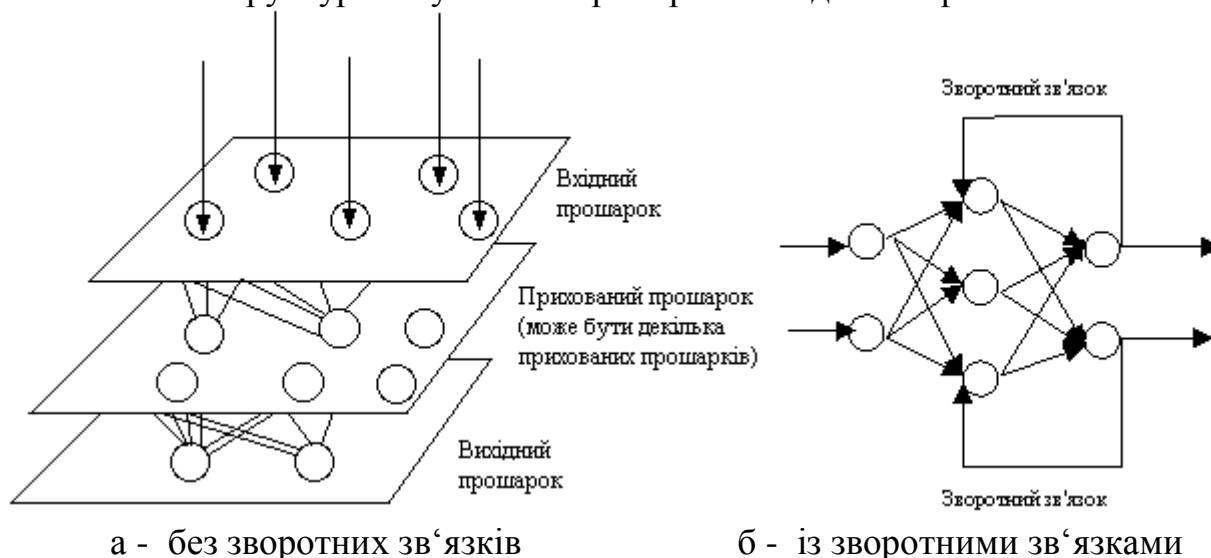


Рис.1.7 Логічна схема перцептрону

Нейромережа

Нейромережа є групуванням штучних нейронів. Це групування обумовлено створенням з'єднаних між собою прошарків

Типові структури штучних нейромереж наведено на рис.1.8 а та б



а - без зворотних зв'язків

б - із зворотними зв'язками

Рис.1.8 Варіанти схем нейромереж

Шлях, яким нейрони з'єднуються між собою має значний вплив на роботу мережі. Більшість пакетів професійної розробки програмного забезпечення дозволяють користувачу додавати, вилучати та керувати з'єднаннями як завгодно. Постійно корегуючи параметри, зв'язки можна робити як збуджуючими так і гальмуючими

Навчання штучної нейронної мережі

Процес навчання може розглядатися як визначення архітектури мережі і налаштування ваг зв'язків для ефективного виконання спеціальної задачі. Нейромережа налаштовує ваги зв'язків по наявній навчальній множині.

Для процесу навчання необхідно 1) мати модель зовнішнього середовища, у якій функціонує нейрона мережа. 2) необхідно визначити, як модифікувати вагові параметри мережі. Алгоритм навчання означає процедуру, в якій використовуються правила навчання для налаштування ваг.

Існують три загальні парадигми навчання: "з вчителем", "без вчителя" (самонавчання) і змішана.

Машина фон Неймана у порівнянні з біологічною системою

Порівняння машина фон Неймана з біологічною системою наведено в табл.1.5.

Таблиця 1.5. Порівняння машина фон Неймана з біологічною системою

Характеристики	Машина фон Неймана	Біологічна нейрона система
Процесор	Складний Високошвидкісний Один чи декілька	Простий Низькошвидкісний Велика кількість
Пам'ять	Відділена від процесора Локалізована Адресація не по змісту	Інтегрована в процесор Розподілена Адресація по змісту
Обчислення	Централізовані Послідовні Збережені програми	Розподілені Паралельні Самонавчання
Надійність	Висока вразливість	Живучість
Спеціалізація	Чисельні і символічні операції	Проблеми сприйняття
Середовище функціонування	Строго визначена Строго обмежена	Погано визначена Без обмежень
Функції	Логічно, через правила, концепції, обчислення	Через зображення, рисунки, керування
Метод навчання	За правилами (дидактично)	За прикладами (сократично)
Застосування	Числова та символічна обробка інформації	Розпізнавання мови Розпізнавання образів Розпізнавання текстів

В теперішній час існує три основних напрямки побудови нейронних обчислювачів

- на базі каскадного з'єднання універсальних RISC або CISC мікропроцесорів фірм Intel, AMD, Sparc, Alpha, Power PC, MIPS.;
- на базі програмовних логічних матриць PLM або процесорів із паралельною обробкою даних на апаратному рівні, наприклад, сигнальних процесорів фірм TMS, ADSP, Motorola;
- на спеціалізованій елементній базі – однокітових процесорів, нейрочипів.

Нейроемулятори

Системи першого напрямку називаються нейроемуляторами. Апаратна їх реалізація базується на використанні універсальних RISC або CISC мікропроцесорів. Нейроемулятори реалізують типові нейрооперації (обчислюють зважену суму, виконують нелінійні перетворення) на програмному рівні.

Цей підхід не використовує переваг паралелізму, що притаманний нейрообчисленням, він орієнтується виключно на здатність нейромереж вирішувати формалізовані задачі.

Переваги таких "віртуальних" нейрокомп'ютерів для відносно невеликих завдань очевидні:

- Не треба витрачатися на нову апаратуру, якщо можна використати комп'ютери загального призначення.
- Користувач не повинен освоювати особливості програмування на спец-процесорах і способи їх сполучення з базовим комп'ютером.
- Універсальні ЕОМ не накладають жодних обмежень на структуру мереж і способи їх навчання, тоді як спец-процесори найчастіше мають обмежений набір «защитих» функцій активації і досягають пікової продуктивності лише на певному колі завдань.

Нейроемулятори використовують готові нейропакети і можуть бути 2 типів:

- Безкоштовні продукти, які, зазвичай, є багат шаровими персептронами з одним або кількома правилами навчання. Виняток становить цілком професійний SNNS (Stuttgart Neural Network Simulator) з великим набором можливостей, що працює, правда, тільки на UNIX-машинах
- Комерційні пакети, що відрізняються від вільно розповсюджуваних великим набором засобів імпорту та передобробки даних, додатковими можливостями з аналізу значущості входів та оптимізації структури мережі. Вартість комерційних емуляторів - масштабу \$ 1000

Прикладами нейроемуляторів є:

- Нейроемулятор FuzzySearch. Програма призначена для демонстрації можливостей мереж Хемінга в розпізнаванні образів.

- Нейроемулятор Neural Network Wizard В ньому реалізовано багатошарову нейронну мережу, що навчається за алгоритмом зворотного поширення похибки. Програма може застосовуватися для аналізу інформації, побудови моделей процесів і прогнозування.
- Нейроемулятор Sharky Neural Network є безкоштовною комп'ютерною програмою, яка здійснює класифікацію за допомогою нейронних мереж
- Нейроемулятор Neuro Color є безкоштовною комп'ютерною програмою, яка здійснює розпізнавання кольорів за допомогою нейронних мереж

Нейроприскорювачі

Нейроприскорювачі поділяються на два класи - “віртуальні”, що вставляються в слот розширення стандартного комп'ютера і “зовнішні”, що з'єднуються з керуючою Host EOM.

Представником 1 класу є нейроприскорювачі на базі програмовних логічних матриць PLM, де алгоритми нейромережі реалізовано апаратно.

До 2 класу відноситься зовнішній нейроприскорювач, на базі каскадного з'єднання сигнальних процесорів. Керуюча Host-EOM реалізована на основі звичайної обчислювальної системи з CISC- або RISC-мікропроцесорами.

Нейроприскорювач другого типу наведений на рис. 1.9

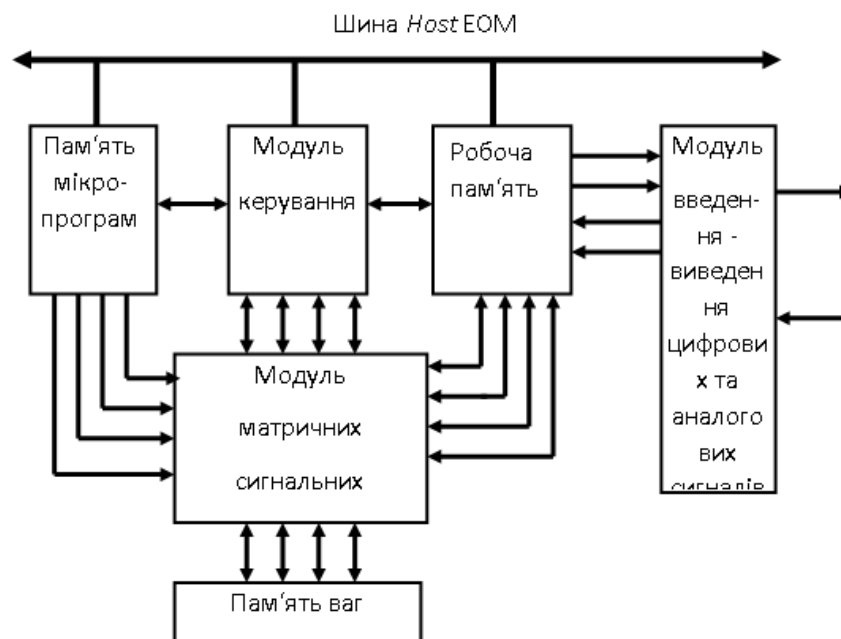


Рис.1.9 Нейроприскорювач на базі каскадного з'єднання сигнальних процесорів

Модуль матричних сигнальних процесорів об'єднує їх між собою відповідно до структури нейромережі. Схема містить також робочу пам'ять, пам'ять програм, модуль введення-виведення сигналів (що включає АЦП, ЦАП і TTL лінії), а також модуль керування, що може бути реалізований на основі спеціалізованого керуючого сигнального процесора, на основі PLM або мати

розподілену структуру, при котрій функції загального керування розподілені між матричними сигнальними процесорами.

Нейрочипи

Елементною базою нейрообчислювачів третього напрямку – саме нейрокомп'ютерів – є нейрочипи.

Нейрочипи бувають цифрові, аналогові і гібридні. Вони також можуть включати схеми настроювання ваг при навчанні, а можуть не мати таких схем і передбачати зовнішнє завантаження ваг.

Прикладом нейрочипа є нНейросигнальний процесор NeuroMatrix NM6403, рис.1.10. Основою є процесорне ядро NeuroMatrixCore (NMC), яке являє собою синтезабільну модель високопродуктивного DSP процесора з архітектурою VLIM / SIMD (мова Verilog). Ядро складається з двох базових блоків: 32-бітного RISC процесора і 64 бітного векторного процесора, що забезпечує виконання векторних операцій над даними змінної розрядності.

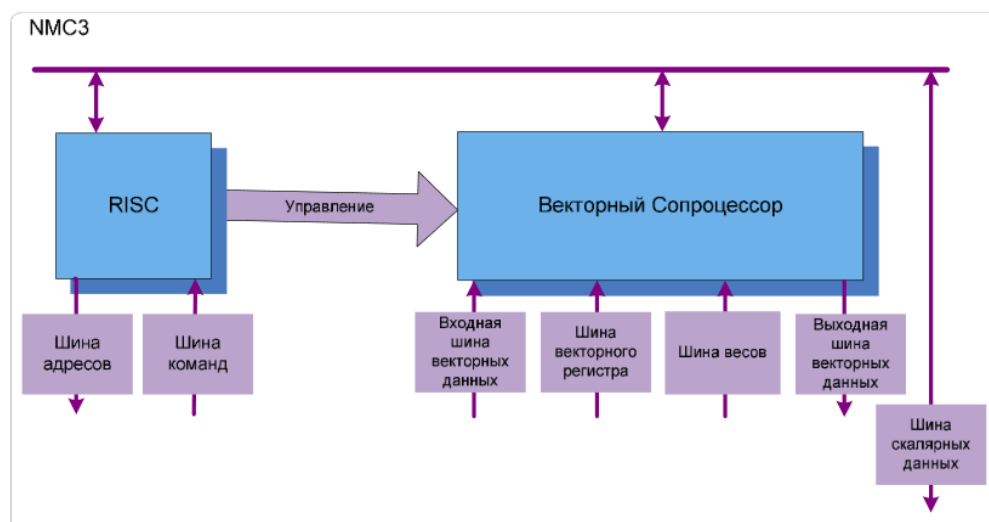


Рис.1.10. Нейросигнальный процессор NeuroMatrix NM6403

Тема 1.6 Квантові комп'ютери

Квантовий комп'ютер— фізичний обчислювальний пристрій, функціонування якого полягає на принципах квантової механіки, зокрема, принципі суперпозиції та явищі квантової запутаності

Квантовий біт або кубіт (q-біт; від quantum bit) — це найменший дворівневий елемент для зберігання інформації в квантовому комп'ютері

В класичному комп'ютері біт є найменшим елементом для зберігання інформації, і він може знаходитися у двох станах 0 та 1. Аналогічно до класичного біта двома можливими станами кубіта є квантовий нуль і квантова одиниця. Різниця між кубітами та бітами полягає у тому, що кубіт може знаходитися в суперпозиції цих станів:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

Де α і β - будь-які комплексні числа, що задовольняють умові нормування:
 $|\alpha|^2 + |\beta|^2 = 1$.

Визначити коефіцієнти α і β можна тільки шляхом вимірювання великої множини однаково приготованих кубітів.

При вимірюванні стану кубіта він переходить в один із базових станів. Імовірність переходу у стан $|0\rangle$ рівна $|\alpha|^2$, а переходу у стан $|1\rangle$ - $|\beta|^2$ відповідно. Геометричне представлення кубіта наведено на рис.1.11.

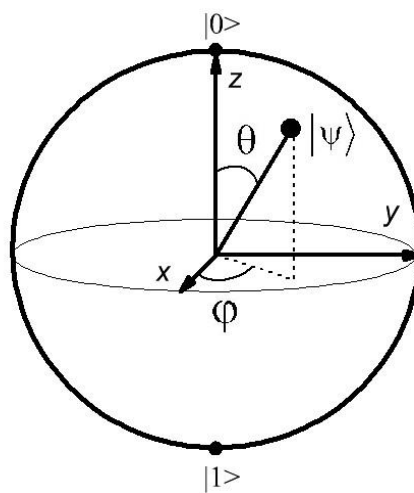


Рис. 1.11 Геометричне представлення кубіта

Геометричне представлення кубіта полягає у відображенні всіх його імовірнісних станів на сфері Блоха. Числа θ і ϕ задають точку на одиничній тривимірній сфері, що називається сферою Блоха

Реалізація кубіта

Кубіт, реалізований на фотонах. У випадку поляризації світлом виникають два стани : стан $|0\rangle$ - горизонтальна поляризація та стан $|1\rangle$ - вертикальна поляризація. У разі якщо за фізичну систему приймається кількість фотонів, то стан $|0\rangle$ - це вакуум, а стан $|1\rangle$ - це наявність фотона.

Кубіт, реалізований на електронах. Якщо за фізичну систему прийняти орієнтацію спіна електрона, то стан $|0\rangle$ - це орієнтація спіна вгору, а стан $|1\rangle$ - це орієнтація спіна вниз. У випадку коли за фізичну систему вибрано кількість електронів стан $|0\rangle$ - це відсутність електронів, а стан $|1\rangle$ - це один електрон.

Кубіт, реалізований на ядрах. За фізичну систему вибираємо ядерний спін. Стан $|0\rangle$ - це орієнтація спіна вгору, а стан $|1\rangle$ - це орієнтація спіна вниз.

Регістр з двох кубітів

Регістр з двох кубітів має чотири базових стани: $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$ та може знаходитися в суперпозиції цих станів. Квантовий стан такого регістру описується вектором:

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle,$$

де $\alpha_{00}, \alpha_{01}, \alpha_{10}, \alpha_{11}$ - амплітуди квантових станів, причому $|\alpha_{00}|^2$ — імовірність виміряти стан $|00\rangle$, $|\alpha_{01}|^2$ — імовірність виміряти стан $|01\rangle$, $|\alpha_{10}|^2$ — імовірність виміряти стан $|10\rangle$, $|\alpha_{11}|^2$ — імовірність виміряти стан $|11\rangle$. Як і для одного кубіта, так і для двох кубітів сума ймовірностей всіх квантових станів буде рівною одиниці

$$|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1.$$

Таким чином, операція над регістром кубітів впливає на всі імовірності будь-якого кубіта з квантового регістру. Така особливість регістру кубітів забезпечує паралелізм обчислень, котрий і надає перевагу квантовим комп'ютерам над класичними.

Квантова запутаність

Квантова запутаність — квантовомеханічне явище, при якому квантовий стан двох або більшої кількості об'єктів повинен описуватися у взаємозв'язку один з одним, навіть якщо окремі об'єкти рознесені в просторі. Внаслідок цього виникають кореляції між спостережуваними фізичними властивостями об'єктів.

Існування явища квантової запутаності дає право говорити про те, що квантовий комп'ютер є набагато потужнішим за класичний. Наприклад, для зберігання N -бітового регістра класичний комп'ютер оперує N класичними бітами. Але аналогічний квантовий регістр описується вектором у 2^N вимірному просторі, тому має бути задано 2^N комплексних коефіцієнтів що дозволяє зберігати в N -кубітовому регістрі одночасно всі 2^N чисел.

Реалізації квантового комп'ютера

У 2007 р. - квантовий комп'ютер із 16 кубітами.

2011 р. - квантовий комп'ютер D-Wave One на чипсеті зі 128 кубітами (рис.1.12)

2017 р. компанія IBM представила прототип квантового комп'ютера з 50 кубіт. В представленому прототипі час когерентності кубіт (час, протягом якого вони можуть залишатись в стані суперпозиції та виконувати корисні обчислення) вдалось збільшити до 90 мікросекунд

9 січня 2018 року на виставці CES 2018 компанія Intel представила чип квантового комп'ютера на 49 кубіт з назвою Tangle Lake. В процесорі використані надпровідні ланцюги

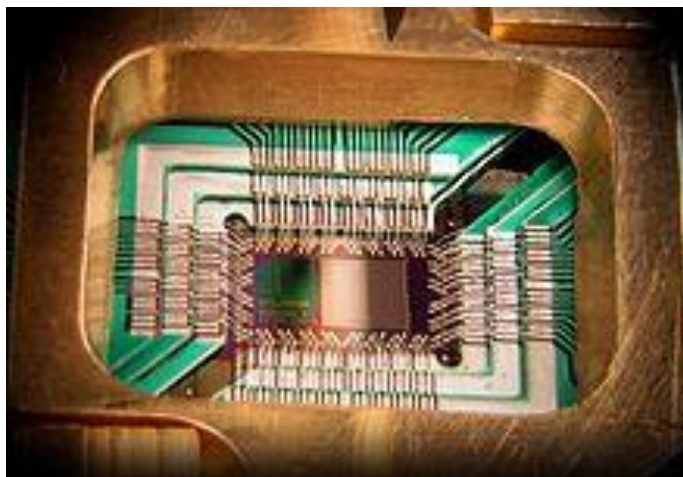


Рис. 1.12 Зразок процесора D-Wave Systems

Тема 1.7 Оптичні комп'ютери

Оптичний або фотонний комп'ютер - це комп'ютер, заснований на використанні оптичних процесорів. На відміну від звичайних комп'ютерів, заснованих на електронних технологіях, в оптичних комп'ютерах операції виконуються шляхом маніпуляції потоками оптичного випромінювання, що дозволяє досягти більшої продуктивності обчислень.

Переваги оптичних технологій

- Принципове підвищення продуктивності
- Можливе зменшення розмірів елементів схем
- Знижена споживана потужність

Оптичні комп'ютери

- Оптичний комп'ютер компанії «Bell Labs»

Перший макет оптичного комп'ютера був створений в 1990 році Аланом Хуаном у (Bell Labs). Процесор другого покоління носив назву «DOC-II» (Digital Optical Computer - цифровий оптичний комп'ютер) і був здатний перевіряти до 80 тис. сторінок тексту в секунду при виконанні команди пошуку слова

- Оптичний комп'ютер компанії «Lenslet»

Компанією «Lenslet» був випущений комерційний оптичний процесор EnLight256. Особливістю його архітектури є те, що в той час, як ядро засноване на оптичних технологіях, всі входи і виходи - електронні. Цей процесор здатний виконувати до 8×10^{12} операцій в секунду. Комп'ютер на базі EnLight256

здатний обробляти 15 відеоканалів стандарту HDTV в режимі реального часу і дозволяє створити новий напрям в голографічному 3D TV

- Фотоний комп'ютер

Інженери з університету Колорадо розробили дослідний зразок 2-ядерного чипа управління фотонного комп'ютера нового покоління, розмір якого становить 3х6 мм. Комп'ютер має світлові порти, через які інформація передається зі швидкістю майже в 300 Гбіт/с. Вчені планують провести ряд додаткових випробувань, і до 2017 року випустити першу партію фотонних процесорів для серійного виробництва. З їх допомогою фахівці збираються в рази збільшити швидкість обробки та обміну даних між дата-центрами, а також збільшити швидкість і час роботи смартфонів, планшетів та інших мобільних гаджетів, без додаткової їх підзарядки

Нанопотоний процесор

Група дослідників з Массачусетського технологічного інституту за участю фахівців компанії Elenion Technologies, Нью-Йорк, і вчених з університету Шербрук, Канада, розробила і реалізувала принципово новий підхід до побудови систем глибинного машинного навчання і штучного інтелекту.

У своїй роботі массачусетські дослідники створили оптичний обчислювальний пристрій, який вони називають програмованим нанопотоним процесором. В середині цього процесора циркулює безліч окремих променів світла, які взаємодіють один з одним за певним алгоритмом і проводять необхідні обчислення.

Оптичний процесор, побудований на такому принципі, може ефективно виконувати операції множення матриць чисел, за рахунок чого програми глибинного машинного навчання зможуть виконуватись набагато швидше і ефективніше. Дослідники прогнозують, що приріст продуктивності в такому випадку буде мінімум дворазовим, а сумарна ефективність системи збільшиться в три рази в порівнянні з традиційними обчислювальними системами.

Для демонстрації можливостей дослідного зразка оптичного процесора, дослідники запрограмували його на реалізацію нейронної мережі, що виконує розпізнавання чотирьох базових голосних звуків. І навіть дослідний зразок нанопотонного процесора, який не відрізняється особливою складністю, зміг відразу забезпечити точність 77 відсотків при виконанні даної задачі. Для порівняння, нейронні мережі, реалізовані на традиційних обчислювальних системах, забезпечують точність близько 90 відсотків при виконанні точно такого ж завдання, однак оптичний процесор практично не споживав енергію порівняно з кількістю енергії, що вимагається для роботи суперкомп'ютера.

Оптичні нейронні мережі – основа надшвидких і надпотужних систем штучного інтелекту

Дослідники з Принстонського університету розробили прототип оптичної нейронної мережі, яка вона здатна вирішити складне математичне рівняння в 1 960 разів швидше, ніж з цим може впоратися процесор середнього рівня

Принстонська система заснована на оптичних “нейронах”, які представляють собою звуження хвилеводів, виготовлених на кремнієвому підставі. Розміри кожного звуження відповідають довжині хвилі світла, на який вони реагують. Світло лазера, що подається в оптичну систему, що складається з безлічі нейронів, проходить через низку операцій над ним і втрачає свою амплітуду (яскравість). Зміни яскравості променю, які можна виміряти з досить високою точністю, і є результатами роботи фотонної нейронної мережі.

оптичні процесори

Оптичні процесори зможуть стати основою інтелектуальних систем управління та інформаційних систем, до яких пред’являються дуже жорсткі вимоги по кількості споживаної енергії. В якості прикладів можна навести приклади систем керування безпілотними апаратами, автономними автомобілями-роботами, мобільні споживчі пристрої та системи обробки сигналів у реальному часі, які є частиною комунікаційних систем.

Тема 1.8 Біокомп’ютери на основі ДНК

Біокомп’ютинг

Біокомп’ютинг (або квазібіологіческая парадигма) - біологічний напрям в штучному інтелекті. зосереджений на розробці і використанні комп’ютерів, які функціонують як живі організми або містять біологічні компоненти, так звані біокомп’ютери.

Родоначалником біологічного напрямку в кібернетиці є У. Мак-Каллок. а також наступні ідеї М. Конрада, які привели до напрямку - біомолекулярні електроніка.

Біомолекулярна електроніка (Нанобіоелектроніка)

У 1974 році А. Авірам і М. Ратнер запропонували] використовувати окремі молекули як елементарну базу електронних пристроїв. Потім М. Конрад запропонував концепцію ферментативного нейрона, засновану на безперервних розподілених середовищах, що обробляють інформацію. Ці ідеї дали початок квазібіологічній парадигмі, яка дозволила практично реалізувати молекулярні нейромережеві пристрої, наприклад, на основі білка бактеріородопсина.

Біокомп’ютинг дозволяє вирішувати складні обчислювальні задачі, організовуючи обчислення за допомогою живих тканин, клітин, вірусів і біомолекул. Часто використовують молекули дезоксирибонуклеїнової кислоти. на основі якого створюють ДНК-комп’ютер. Крім ДНК, як біопроцесор можуть

використовуватися також білкові молекули і біологічні мембрани. Наприклад, на основі бактеріородопсин-плівок створюють молекулярні моделі перцептронів.

ДНК комп'ютер

ДНК-комп'ютер - обчислювальна система, яка використовує обчислювальні можливості молекул ДНК.

У 1994 році Леонард Адлеман, професор університету Південної Каліфорнії, продемонстрував, що за допомогою пробірки з ДНК можна досить ефективно вирішувати класичне комбінаторне «завдання про комівояжера» (найкоротший маршрут обходу вершин графа). Класичні комп'ютерні архітектури вимагають множини обчислень з випробуванням кожного варіанта.

Метод ДНК дозволяє відразу згенерувати всі можливі варіанти рішень за допомогою відомих біохімічних реакцій. Потім можливо швидко відфільтрувати саме ту молекулу-нить, в якій закодована потрібна відповідь.

У 2002 році дослідники з Інституту Вейцмана в Реховоті, Ізраїль, представили програмовану молекулярну обчислювальну машину, що складається з ферментів і молекул ДНК. 28 квітня 2004 року, Ехуд Шапіро, Яков Бененсон, Біньямін Гіл, Урі Бен-Дор і Рифка Адар з Інституту Вейцмана повідомили в журналі «Nature» про створення ДНК-комп'ютера з модулем введення-виведення даних.

У січні 2013 року дослідники змогли записати в ДНК-коді кілька фотографій JPEG, набір шекспірівських сонетів і звуковий файл.

У березні 2013 року дослідники створили транскриптор (біологічний транзистор).

Принцип роботи ДНК комп'ютера наступний:

Нитки ДНК мають в своєму складі чотири азотистих основи: цитозин, гуанін, аденін, тимін. Їх послідовність кодує інформацію. За допомогою ферментів цю інформацію можна змінювати: полімерази добудовують ланцюжки ДНК, а нуклеази їх розрізають і вкорочують. Деякі ферменти здатні розрізати і з'єднувати ланцюга ДНК в місцях, що вказуються іншими ферментами - лігазами. Таким чином, ДНК-комп'ютери можуть зберігати і обробляти інформацію. Також, хімічні реакції на різних частинах молекул проходять незалежно, паралельно, що забезпечує високу швидкість обчислень

РОЗДІЛ 2 Характеристики ARM і Cortex процесорів

Тема 2.1 Передумови створення ARM. Процесори Cortex

Переваги RISC – процесорів

Аналіз коду програм , генерованого компіляторами мов високого рівня, показав , що найчастіше використовується тільки обмежений набір простих команд таких форматів, як «регістр», «регістр → регістр» і «регістр ↔ пам'ять». Компілятори неефективно транслюють складні команди, створюючи при цьому надлишковий програмний код. Це спостереження сприяло формуванню концепції процесорів із скороченим набором команд, так званих RISC – процесорів (RISC – Reduced Instruction Set Computer). Це призводить до зменшення розміру кристалу. Звернення до пам'яті виконується за допомогою спеціальних команд завантаження регістра й запису. Простота структури і невеликий набір команд дозволяє реалізувати повністю їх апаратне виконання й ефективний конвеєр при невеликому обсязі обладнання. Арифметику RISC - процесорів відрізняє високий ступінь дроблення конвеєра. Цей прийом дозволяє збільшити тактову частоту а, отже , і продуктивність комп'ютера: чим елементарніші дії виконуються в кожній фазі роботи конвеєра , тим вище частота його роботи. RISC – процесори з самого початку орієнтовані на реалізацію всіх можливостей для прискорення арифметичних операцій, тому їхні конвеєри мають більш високою швидкість, ніж у CISC - процесорах. Відповідно до цього RISC – процесори у 2–4 рази швидше за CISC – процесори при тій самій тактовій частоті та звичайною системою команд і продуктивніші, незважаючи на більший обсяг програм (рис.2.1).

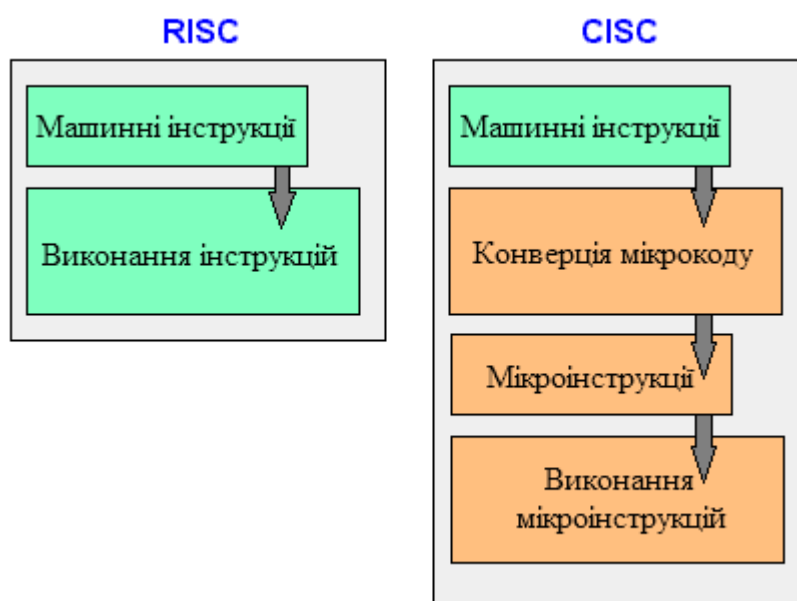


Рис. 2.1. Виконання інструкцій у RISC та CISC процесорах

З цього випливають чотири основних принципу RISC:

1. Будь-яка операція має бути виконаною за 1 такт незалежно від її типу.
2. Система команд має містити мінімальну кількість команд, що мають найчастіше застосування з однаковою довжиною.
3. Операції обробки даних реалізуються у форматі «регістр-регістр» (операнди обираються з оперативних регістрів процесора й результат записується також у регістр, а обмін між оперативними регістрами й пам'яттю здійснюється за допомогою команд завантаження/запису.
4. Склад команд має бути зручним для компіляції операндів мов високого рівня.

Отже, характерні особливості RISC – процесорів:

1. Фіксована довжина машинних інструкцій і простий формат команди.
2. Спеціалізовані команди для операцій з пам'яттю: читання або запису. Операції виду «Читання Модифікація Запис» відсутні. Будь-які операції «Зміна» виконуються лише над вмістом регістрів, а результат поміщається тільки в регістр.
3. Велика кількість регістрів загального призначення (32 і більше).
4. Відсутність підтримки операцій виду «Зміна» над укороченими типами даних таких як байт і 16-ти бітне слово. Так, наприклад, система команд DEC Alpha містила лише операції над 64-бітними словами, і вимагала розробки та подальшого виклику процедур для виконання операцій над байтами, 16-ти і 32-ти бітними словами.
5. Відсутність мікропрограм всередині самого процесора. Те, що в CISC - процесорі виконується за допомогою мікропрограм, то в RISC - процесорі виконується як звичайний машинний код, не відрізняється принципово від коду ядра операційної системи та додатків.

Характеристики ARM процесорів

Саме необхідність в процесорі підвищеної потужності, здатному працювати з графічним інтерфейсом користувача, призвела на початку 80-х років минулого сторіччя британську компанію Acorn Computers до думки про необхідність відмови від готових, але малопотужних рішень, пропонованих партнерами MOS Technology і Motorola, і запуску розробки нового власного процесора.

Учасники проекту BBC Micro створили для Acorn мікропроцесор на архітектурі RISC. RISC-архітектура пропонувала оптимізацію обчислювального процесу за рахунок реалізації складних функцій не за допомогою єдиної комплексної команди, як це робилося в CISC, а за допомогою набору більш простих команд. При такому підході арифметико-логічний пристрій істотно спрощується, що дозволяє додати в схему процесора більшу кількість регістрів. Збільшення кількості регістрів знижує необхідність частого звернення до повільною оперативної пам'яті.

У фірмі ARM обрали вектор розвитку мікропроцесорів спрямований у бік технологій ASIC і ASSP.

Технологія ASSP (Application-specific standard products) передбачає розробку простих, але в той же час універсальних по застосуванню компонентів - наприклад, апаратних декодерів звуку і відео.

Технологія ASIC (Application-specific integrated circuit) на противагу ASSP передбачає створення інтегральних мікросхем, що спеціалізуються на вирішенні деякого обмеженого кола завдань. До ASIC-рішень можна віднести роутери, мобільні телефони й ігрові консолі.

В основі ASIC-системи лежить процесорне ядро, контролери пам'яті і периферійних пристроїв.

Отже, завдяки такій самодостатності та поєднанню технологій ASIC та ASSP можна за дуже короткий термін можна створити новий пристрій довільної конфігурації, адаптований саме для використання під конкретно визначені задачі

Вперше 32-розрядна архітектура ARM першої версії (ARMv1) була застосовано в процесорі *ARM1* і використовувалася як заміна малопотужним CISC-процесорам в комп'ютері BBC Micro Група завершила розробку зразків під назвою ARM1 у квітні 1985 року, і першу «реальну» робочу систему ARM2 вже в наступному році.

ARM2 архітектура характеризувалася:

- 32- розрядною шиною даних,
- 26-розрядним (64 Мбайт) адресним простором;
- шістнадцятью 32х розрядних регістрів.

ARM2 є найпростішим із використовуваних 32- бітних мікропроцесорів у світі, робота якого забезпечена лише 30 тис. транзисторами (для порівняння : рання модель Motorola 68000 містила близько 70тис. транзисторів). Така простота походила від відсутності мікропрограм (що становить приблизно від однієї чверті до однієї третини процесору 68000), ARM2 не містив кеш-пам'яті. Це призвело до нижчої швидкості виконання, ніж у процесора Intel 80286. Наступний процесор ARM3, був випущений із 4Кбайтою кеш-пам'яттю, що поліпшило показники.

Найуспішнішої реалізації набув процесор ARM7TDMI, який присутній у майже кожному побутовому приладі, так як став основою для сотень мільйонів мікроконтролерів. ARM Holdings розробляє набір інструкцій та архітектуру процесорів , проте не виробляє самих кристалів. Компанія періодично випускає оновлення для своїх ядер.

Еволюція серій процесорів та їх ядра ARM наведений у табл.2.1.

Таблиця 2.1 Характеристики ARM процесорів

Серія	Версія архітектур и	Ядро	Модернізація	Кеш Кб	f, МГц
ARM1	ARMv1	ARM1	базовий	-	8
ARM2	ARMv2	ARM2	Додана інструкція множення MUL		
	ARMv2a	ARM250	Інтегрований МЕМС (Блок керування пам'яттю), Графічний процесор та процесор вводу-виводу. Архітектура 2a додала SWP і SWPB (своп) інструкції.	-	12
ARM3	ARMv2a	ARM2a	Додано процесорний кеш	4	25
ARM6	ARMv3	ARM60	32 бітна шина даних	-	12
		ARM600	Співпроцесорна шина для FPA10 розрахунків.	4	33
		ARM610	Відсутня співпроцесорна шина	4	20
ARM7	ARMv3	ARM700	-	8	40
		ARM710	-	8	40
		ARM710a	-	8	40
		ARM7100	Інтегрована система на кристалі	8	18
		ARM7500	Інтегрована система на кристалі	4	40
		ARM7500 FE	Інтегрована система на кристалі. Додано FPA та EDO.	4	56
ARM7TD MI	ARMv4T	ARM7TD MI(-S)	Конвеєр з трьох етапів, thumb.	-	16,8
		ARM710T	-	8	40
		ARM720T	-	8	59,8
		ARM740T	-	-	59,8
	ARMv5TEJ	ARM7EJ-S	Конвеєр з 5 етапів, вдосконалені інструкції з обробки цифрових сигналів, Jazelle DBX.	-	59,8
Strong ARM	ARMv4	SA-110	-	16	203
		SA-1110	-	16	233
ARM8	ARMv4	ARM810	Конвеєр з 5 етапів, пам'ять із подвійною пропускнуою шиною	8	72
ARM9 TDMI	ARMv4T	ARM9 TDMI	Конвеєр з 5 етапів	-	180
		ARM920T	-	16	180
		ARM922T	-	8	180
		ARM940T	Блок захисту пам'яті.	4	180
ARM9E	ARMv5TE	ARM946 E-S	Вдосконалені інструкції обробки цифрових сигналів.	Є	220
		ARM966 E-S		-	220

		ARM968 E-S		-	220
	ARMv5TEJ	ARM926 EJ-S	Вдосконалені інструкції обробки цифрових сигналів. Jazelle DBX.	є	220
	ARMv5TE	ARM996 HS	Самотактовий процесор, вдосконалені інструкції обробки цифрових сигналів	-	220
ARM10E	ARMv5TE	ARM1020 E	Конвеєр шести рівнів, вдосконалені інструкції обробки цифрових сигналів, VFP.	32	220
		ARM1022 E	VFP.	16	220
	ARMv5TEJ	ARM1026 EJ-S	Вдосконалені інструкції обробки цифрових сигналів. Jazelle DBX.	є	220
XSCALE	ARMv5TE	80200 IOP310 OP315	-	є	220
		80219	-	є	400/600
		IOP321	-	є	600
		IOP33x	-	є	600
		IOP34x	1-2 ядра, прискорення RAID.	32	600
		PXA210	Конвеєр семи рівнів. Прикладний процесор.	32	400
		PXA255	-	32	400
		PXA26x	-	-	400/624
		PXA27x	Прикладний процесор.	32	624
		PXA800	-	32	624
		Monahans	-	32	1250
		PXA900	-		
		IXC1100	Процесор панелі керування.		
		IXP2400 IXP2800			
		IXP2850			
		IXP2325 IXP2350			
		IXP42x			
		IXP460 IXP465			
ARM11	ARMv6	ARM1136 J(F)-S	SIMD, Jazelle DBX, (VFP), Конвеєр восьми етапів.	є	532/665
	ARMv6T2	ARM1156 T2(F)-S	SIMD, Thumb-2, (VFP), Конвеєр дев'яти етапів.	є	665
	ARMv6KZ	ARM1176 JZ(F)-S	SIMD, Jazelle DBX, (VFP).	є	665
	ARMv6K	ARM11	1-4 ядра SMP, SIMD, Jazelle DBX,	є	665

		MP Core	(VFP).		
Cortex	ARMv7-A	Cortex-A8	Прикладний процесор, VFP, NEON, Jazelle RTC, Thumb-2, Конвеєр з тринадцяти етапів.	є	600/1000
		Cortex-A9	Прикладний процесор, VFP, NEON, Jazelle RTC та DBX, Thumb-2, багатократний, суперскалярний конвеєр з восьми етапів, виконання поза порядком.	є	600/1000
		Cortex-A9 MPCore	Те, що й Cortex-A9, ядро SMP.	є	600/1000
	ARMv7-R	Cortex-R4	Поглиблено-вбудований процесор FPU.	є	
	ARMv7-M	Cortex-M3	Профіль мікроконтролеру, Thumb-2	-	100
	ARMv6-M	Cortex-M0	Профіль мікроконтролеру, Thumb-2 (16-ти бітні команди та BL, MRS, MSR, ISB, DSB, DMB).	-	
		Cortex-M1	Для застосування в FPGA, профіль мікроконтролеру, Thumb-2 (BL, MRS, MSR, ISB, DSB, DMB).	-	170

Процесори CORTEX

На відміну від інших ядер ARM, сімейство Cortex являє собою закінчене процесорне ядро зі стандартним CPU і системною архітектурою. Сімейство Cortex має три основних профіля: профіль A для високопродуктивних додатків, R - для додатків реального часу, M - для бюджетних додатків.

Процесори ARM Cortex-M є сімейством енергоефективних, простий у використанні процесорів, призначених для допомоги розробникам у задоволенні потреб створення нових пристроїв для вирішення сучасних завдань електроніки. Cortex-M мають більше можливостей на відміну від попередніх серій за нижчою ціною, мають можливість підключення до багатьох систем, володіють здатністю використання повторного коду та підвищення ефективності використання енергії.

Сімейство Cortex-M оптимізовано для використання з дорогими, потужними й чутливими MCU та для інших сигнальних пристроїв задля створення кінцевих додатків, таких як smart вимірювання, пристрої введення/виведення інформації, автомобільних і промислових систем управління, побутової техніки, споживчих товарів і медичних приладів. На рис.2.2 показаний напрям розвитку сімейства Cortex-M. З цього рисунку видно, що збільшення розмірів кристалу та модернізація його ядра супроводжується зменшенням вартості.

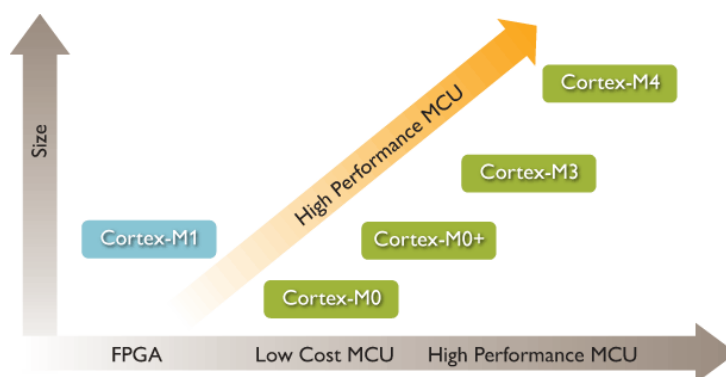


Рис. 2.2

ARM Cortex Microcontroller Software Interface Standard (CMSIS) є програмним інтерфейсом для роботи з периферійними модулями ядра з використанням мови Сі при роботі з регістрами. Його використання передбачено у інтегрованих програмних середовищах таких як Keil, CoCoX. Детальніше про CMSIS буде розглянуто пізніше.

Серія Cortex Cortex-M0

Процесор ARM Cortex-M0 є найменшим процесором ARM серед усіх доступних. Виключно невелика площа кремнію, низьке енергоспоживання і мінімальний код дозволяє розробникам досягти продуктивності 32-ти розрядного за ціною 8-ми бітного. Завдяки ultra-low gate count особливості процесор Cortex-M0 часто використовується у аналогових і змішаних сигнальних пристроях. Структурна схема зображена на рис 2.3.

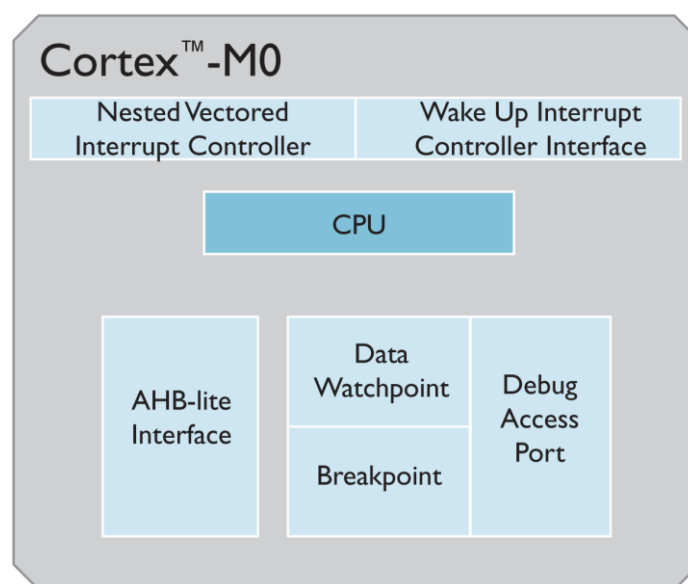


Рис. 2.3 Структурна схема Cortex-M0

Відповідно до рисунку, процесор Cortex-M0 складається з наступних блоків:

- NVIC контролер;

- WVIC інтерфейс;
- Ядро;
- АНВ інтерфейс;
- Debug пристрій.

Спеціалізація Cortex-M0 процесора у пристроях, де необхідна найбільша енергоефективність. На мінімальних налаштуваннях споживання енергії встановлено на рівні 16 мкВт/МГц.

Швидкодія процесора пов'язана з наявністю всього 56 інструкцій. Завдяки архітектурі Cortex-M0 можливо оперувати інструкціями за допомогою мови Cі. Є можливість достатньо просто порахувати час виконання кожної команди.

Cortex-M0+

Відмінністю від Cortex-M0, є додавання до структури схеми наступних блоків (рис 2.4..) :

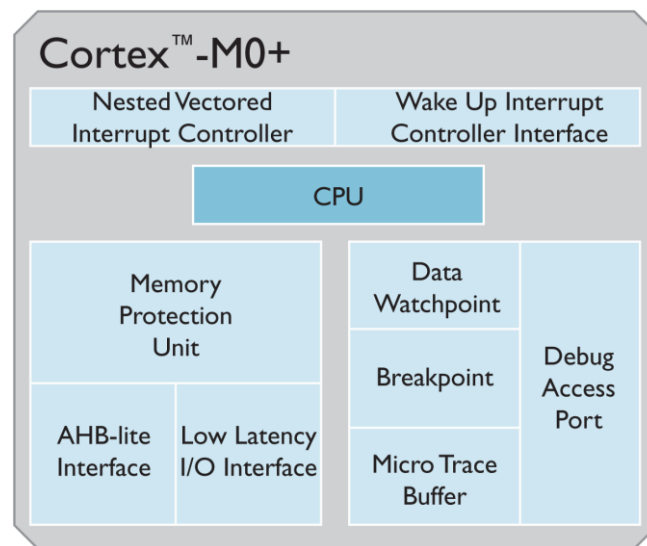


Рис. 2.4. Структурна схема **Cortex-M0+**

- Блок захисту пам'яті;
- швидкодіючий інтерфейс введення/виведення;
- Буфер мікротрасування у блоці відладки.

Незважаючи на додавання нових елементів, енергоспоживання вдалося зменшити на 6,2 мкВт/МГц і складає 9,8 мкВт/МГц.

Cortex-M1

Процесор Cortex-M1 є першим ARM процесором, який був спроектований спеціально для впровадження до FPGA (Field-Programmable Gate Array). Більшість FPGA пристроїв містить процесор Cortex-M1, що дає змогу широкого використання сучасних інструментів FPGA для оптимальної розробки. Процесор Cortex-M1 дозволяє OEM-виробникам досягти значної економії коштів за рахунок раціоналізації програмного забезпечення.

Основні блоки даного процесора показані на рис. 2.5.

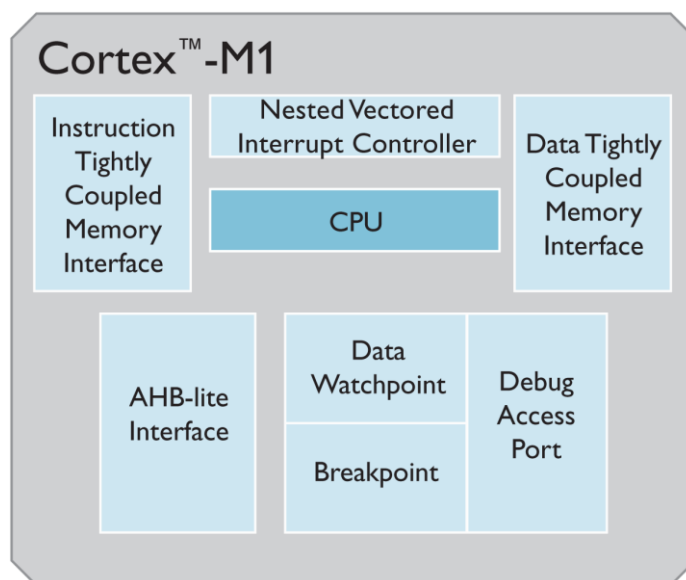


Рис. 2.5. Структурна схема **Cortex-M1**

Переваги використання ARM Cortex-M1 в FPGA:

- Стандартна архітектура процесора;
- незалежність від постачальника – даний процесор підтримує всіх FPGA виробників;
- проста міграція шляху з FPGA до ASIC;
- проста міграція до новіших FPGA пристроїв.

Інші області застосування: засоби комунікації, трансляції; автомобілі, військова техніка, повітряна техніка тощо.

Cortex-M3

Процесор ARM Cortex-M3 є 32-х бітним процесором для додатків, яким необхідна точна real-time інтеграція. Використовується розробниками у широкому колі пристроїв, таких як мікроконтролери, автоматизовані системи, промислові системи керування, бездротовий зв'язок, сенсори тощо. Процесор забезпечує високу обчислювальну продуктивність і негайну реакцію системи на певні події маючи при цьому низьке споживання енергії.

Налаштування Cortex-M3 не викликає складностей тим самим надає широкий спектр для застосування у пристроях, які вимагають захисту пам'яті та складного трасування та які потребують мінімальної площі друкованої плати.

Структурні блоки, з яких складається Cortex-M3 зображені на рис.2.6

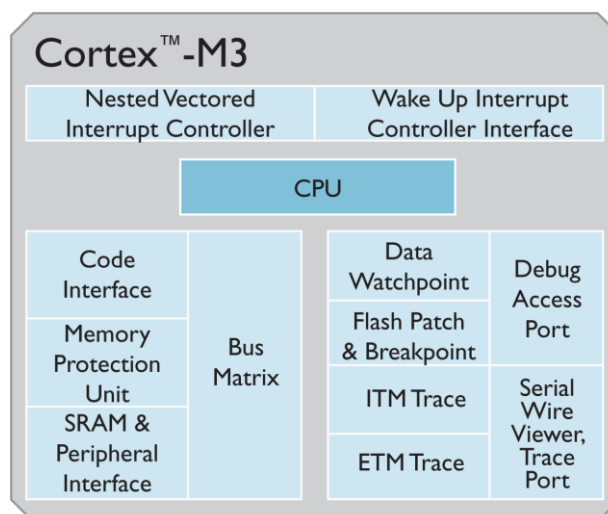


Рис. 2.6. Структурна схема **Cortex-M3**

У порівнянні з попередніми процесорами Cortex-M, до Cortex-M3 входять:

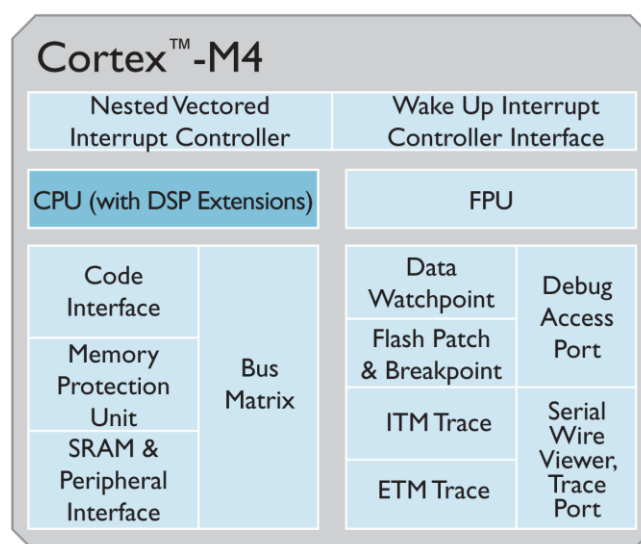
- Шинна матриця, яка складається з декількох шин;
- блок пам'яті SRAM та периферійного інтерфейсу;
- інтерфейс для зручності написання коду;
- спеціальний порт для трасування й відладки.

Найширше застосування Cortex-M3 знайшов у мікроконтролерних системах завдяки своїй гнучкості в налаштуваннях, компактності коду (застосовано технологію Thumb-2 інструкцій, що дозволяє виконувати складні інструкції заміною на більш прості) та енергоефективності. У даному процесорі присутні функції сплячого та активного режиму, який також має низьке споживання енергії. Процесор містить NVIC, що підтримує налаштування до 240 переривань та модуль USB OTG.

Cortex-M4

Сучасною серією Cortex-M процесорів є процесор Cortex-M4. Він спеціально розроблений для задоволення потреб ринку у пристроях з обробкою сигналів управління, а саме цифрових, маючи при цьому високу продуктивність за низьким споживанням енергії, вигідну ціну. Відповідно до цього ARM розробило Cortex-M4 для ефективного використання у пристроях керування двигуном, в автомобілях, керування живленням, відтворення звукової та відео інформації.

Фактично Cortex-M3 та Cortex-M4 мають різницю лише в ядрі (рис. 2.7.), тому більшість переваг Cortex-M3 зберігаються й у Cortex-M4.

Рис. 2.7. Структурна схема **Cortex-M4**

Cortex-M4 має CPU із модулем DSP (Digital Signal Processing) та співпроцесор FPU (Floating Point Unit). DSP дає можливість оброблювати цифрові сигнали, а FPU виконувати точні та швидкі обчислення чисел із плаваючою комою. Оптимізована робота арифметики. SIMD. Відповідно до цього Cortex-M4 спеціалізується на пристроях із цифровою обробкою інформації.

Мікроконтролер STM32F407VG

Нова серія мікроконтролерів STM32F4 є розширенням платформи STM32 заснованим на останній версії ядра ARM Cortex-M4.

У новій серії STM32 F4 додано DSP інструкції, що виконуються за один такт, що покращують обчислювальну здатність для особливо вимогливих в цьому плані пристроїв, таких як медичне обладнання, управління двигунами і охоронне устаткування. Забезпечуючи апаратну (повний збіг пінів) і програмну сумісність з серією STM32 F2, і в той же час, маючи більший обсяг SRAM, підвищену продуктивність і кращу завадостійкість периферії, серія F4 дозволить розробникам поліпшити кінцевий продукт, якщо їм необхідно більше обсягу пам'яті, продуктивності або особливостей периферії.

STM32F407 має наступні переваги перед мікроконтролерами цієї серії: розширена периферія, на додаток до периферії STM32F405 доданий другий USB-OTG інтерфейс, інтегрований Ethernet MAC 10/100 з підтримкою MII і RMIІ, від 8 до 14-ти розрядний інтерфейс камери, що дозволяє зробити з'єднання з КМОП-камерами при роботі на швидкості до 67.2 Мбайт/с. Мікроконтролери STM32F407 доступні в чотирьох типах корпусів - LQFP100, LQFP144, LQFP / BGA176, і Flash-пам'яттю від 512 Кбайт до 1 Мбайта.

Структурна схема ARM Cortex-M4 наведена на рис.2.8.

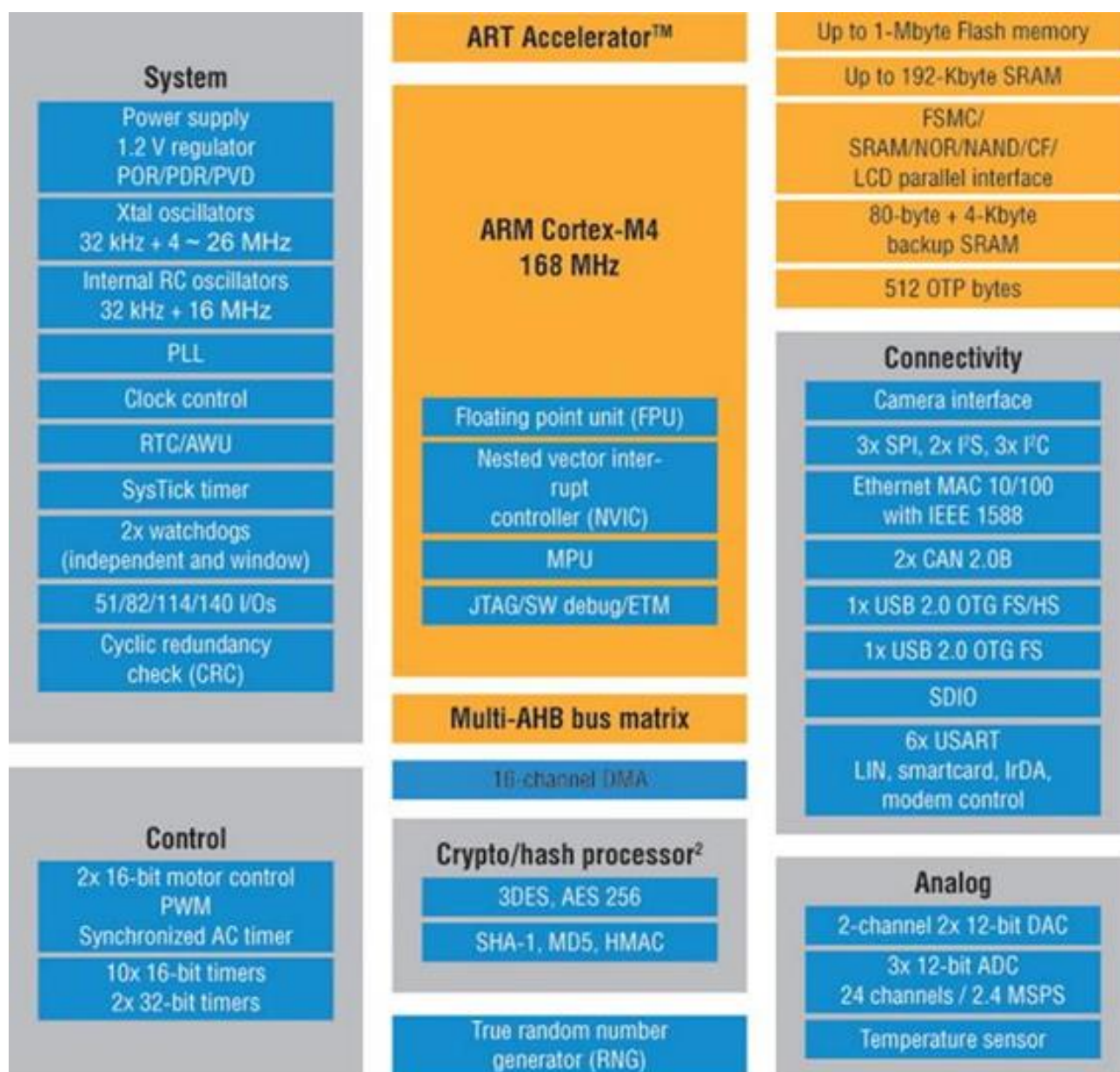


Рис. 2.8. Структурна схема мікроконтролера STM32F407VG

Схема містить

Блок ядра (ARMCortex-M4 168 MHz):

Ядро ARM 32-bit Cortex-M4 CPU;

Частота тактування 168МГц, 210 DMIPS / 1.25 DMIPS / МГц (Dhrystone 2.1);

Підтримка DSP-інструкцій

Блок ART прискорювача (ART Accelerator);

Блок високопродуктивної АНВ-матриці шин (Multi-AHB bus matrix)

Блок пам'яті (Memory):

До 1 МБайт Flash-пам'яті;

До 196 кбайт SRAM-пам'яті;

Контролер SDIO (карти SD, SDIO, MMC, CE-ATA);

FSMC-контролер (Compact Flash, SRAM, PSRAM, NOR, NAND і LCD 8080/6800);

Системний блок (System):

Напруга живлення 1,8 ... 3,6 (POR, PDR, PVD і BOR);

Внутрішні RC-генератори на 16МГц і 32кГц (для RTC);

Зовнішнє джерело тактування 4 ... 26МГц і для RTC - 32,768кГц;

Апаратне обчислення CRC, 96-бітний унікальний ID;

Зовнішній осцилятор 32kHz +16 MHz

Xtal осцилятори 32kHz

Регулятор POR/PDR/PVD

51/82/114/140 I/Os

Clock control

Блок роботи з аналоговими сигналами (Analog):

Три 12-бітних АЦП на 24 вхідних каналу (швидкість до 7,2 мегасемплів, температурний датчик);

Два 12-бітових ЦАП;

Блок DMA-контролера на 16 потоків з підтримкою пакетної передачі (16-channelDMA);**Блок управління (Control):**

17 таймерів (16 і 32 розрядні);

Два сторожових таймера (WDG і IWDG);

мікроконтролер STM32F407VG

Блок комунікації (Connectivity)

Комунікаційні інтерфейси: I2C, USART (ISO 7816, LIN, IrDA), SPI, I2S; CAN (2,0 В Active);

USB 2.0 FS / HS OTG;

10/100 Ethernet MAC (IEEE 1588v2, MII / RMI);

Інтерфейс цифрової камери (8/10/12/14-бітові режими);

Апаратний генератор випадкових чисел;

Блок криптопроцесора (Crypto/hash processor)

Модуль шифрування AES 128, 192, 256, Triple DES, HASH (MD5, SHA-1), HMAC;

Розширений температурний діапазон $-40 \dots 105^{\circ}\text{C}$

Зовнішній вигляд STM32F407VG та STM32F407VG з TFT-драйвером наведено на рис. 2.9 а та б.



Рис. 2.9.Зовнішній вигляд STM32F407VG (а) та STM32F407VG з TFT-драйвером (б)

Тема 2.2. Шини ARM-процесорів Організація пам'яті у мікроконтролерах

Шинна матриця

Всю периферію на чіпі з ARM-ядром об'єднує шина AMBA (Advanced Microcontroller Bus Architecture).

Шина AMBA, в свою чергу, ділиться на системну шину (АНВ Advanced High - performance Bus) і шину периферійних пристроїв (АРВ). Завданням АНВ є забезпечення високошвидкісного обміну між ядром ARM і контролерами пам'яті і LCD-екрану, в той час як мета АРВ - зниження енергоспоживання при роботі з підключеною периферією.

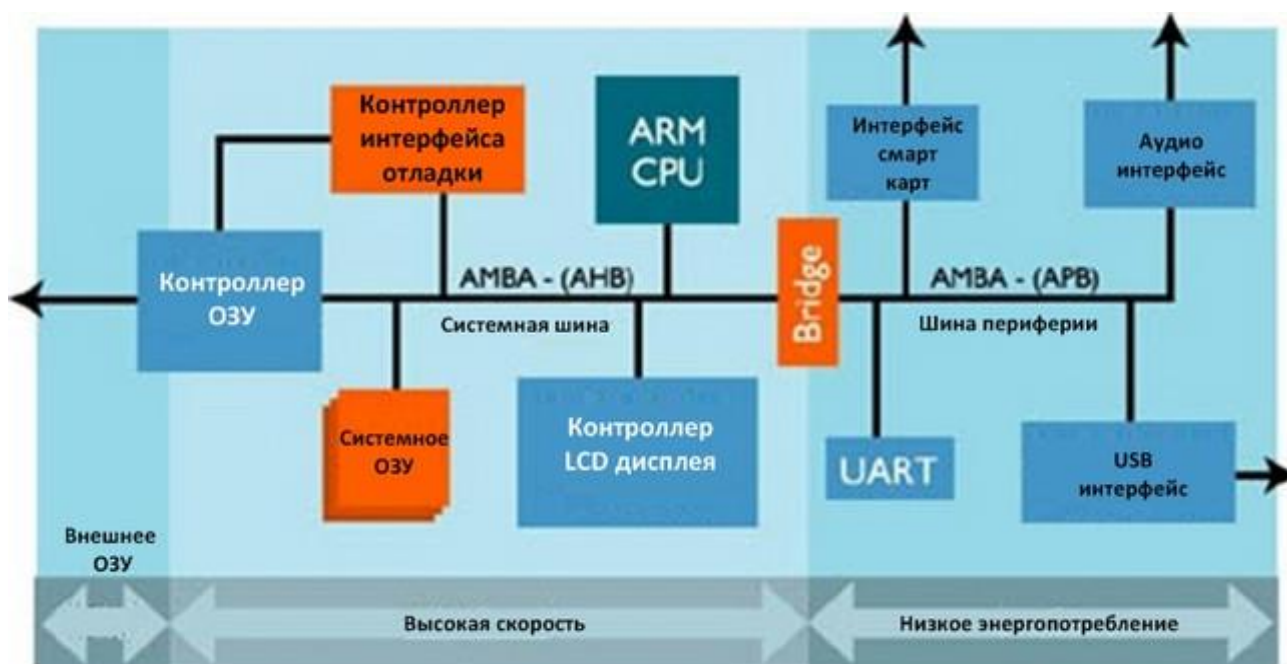


Рис.2.10 Шина AMBA (Advanced Microcontroller Bus Architecture).

АНВ - це відносно нова специфікація для більш продуктивних шин, шини цього типу використовують, в основному, для зв'язку високошвидкісних внутрішніх компонентів, а шину APB, як більш повільну - для периферії типу GPIO, UART та ін.

Системна шина і шина даних підключаються до зовнішнього мікроконтролеру через набір високошвидкісних шин, яка називається матрицею шин. Вона утворює декілька паралельних з'єднань між шинами Cortex. Якщо два шинних майстра роблять спробу доступу до одного і того ж УВВ, то вступить в дію внутрішній арбітр, який розв'яже конфлікт, надавши доступ до шини того, хто має найвищий пріоритет. Однак, завдяки тісному взаємозв'язку блоків ПДП з ЦПУ Cortex, необхідність арбітражу в багатьох випадках виключається

Сьогодні шини стали складніше, їх число збільшилося на порядок. Роздільні шини дають можливість працювати з декількома пристроями одночасно, причому на різних швидкостях: у кожної шини може бути своя швидкість. Можна гнучко управляти енергоспоживанням, відключаючи невикористовувані пристрої й цілі шини. Для управління всією цією кількістю шин знадобилося вводити спеціальний контролер, керуючий обміном даними між шинами і процесором.

Шинна матриця - це розвиток ідеї простого контролера шини: тут шини з'єднані так, що пристрої можуть взаємодіяти безпосередньо, не через ядро. Також вона керує доступом до не-вирівняних даних (адреси яких не кратні 4, як

прийнято в 32-бітових архітектурах) і атомарним доступом до окремих бітів у спеціально виділеному діапазоні (технологія bit-banding)

У мікропроцесорах STM32F405xx / 07xx та STM32F415xx / 17xx, основною шиною є 32-бітова багатопотокова АНВ матриця шин, яка з'єднує вісім майстер - шин та сім слейв - шин. (рис. 2.11)

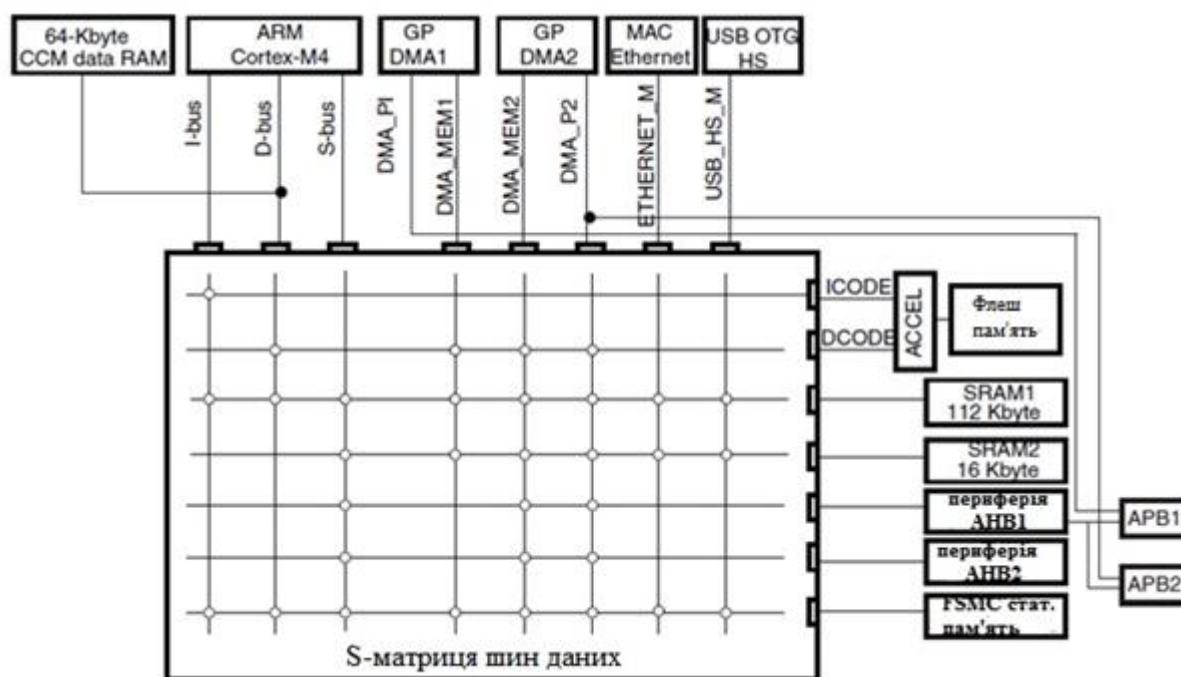


Рис. 2.11 Матриця шин

Майстер - шини наступні:

- Шина ядра: I-шина, D-шина та S-шина;
- шина пам'яті DMA-1;
- шина пам'яті DMA-2;
- периферійна шина DMA-2;
- Ethernet - шина DMA;
- USB OTG HS DMA шина;

Слейв - шини складаються з:

- ICode-шини вбудованої флеш-пам'яті;
- DCode-шини вбудованої флеш-пам'яті;
- шини головної вбудованої SRAM1 (112kB);
- шини допоміжної вбудованої SRAM2 (16kB);
- шини AHB1 периферії;

- шини ANB2 периферії;
- шини FSMC.

Організація пам'яті у мікроконтролерах STM32

Програмна пам'ять, пам'ять даних, регістри і порти введення / виводу організовані в межах лінійних 4 Гбайт адресного простору. Адресний простір пам'яті ділиться на 8 основних блоків, кожен з 512 Мб.

На рис. 2.12 представлена узагальнена карта пам'яті для мікроконтролерів STM32F40xx.



Рис. 2.12 Карта пам'яті STM32F40xx

Перший 1 Гб пам'яті розділений рівномірно між областю коду програми і областю статичного ОЗП. Простір коду програми оптимізовано для роботи з шиною I-Code. Аналогічно, простір статичного ОЗП доступний через шину D-code. Незважаючи на те, що в області статичного ОЗП підтримується завантаження і виконання інструкцій, їх вибірка здійснюється через системну шину, що вимагає додаткового стану очікування. Таким чином, виконання коду програми з статичного ОЗП буде більш повільним, ніж з вбудованої Flash пам'яті, розташованої в області коду програми. Наступні 0.5 Гб пам'яті - область вбудованих УВВ. У цій області знаходяться все, що надаються користувачеві виробником мікроконтролера УВВ. Перший 1 Мб в областях статичного ОЗП і

У STM32F405xx / 07xx і STM32F415xx / 17xx заплановано вбудований статичний ОЗП має доступ як до байтів, півслів (16 біт) або повних слів (32 біта).

Статичний ОЗП (рис. 2.13) ділиться на три блоки:

- 1) SRAM1 і SRAM2 відображається в адресі 0x2000 0000 і доступна для всіх АНВ майстрів.
- 2) SRAM3 (доступна на STM32F42xxx і STM32F43xxx) відображається за адресою 0x2002 0000 і доступна для всіх майстрів АНВ.
- 3) CCM (ядро, поєднане з пам'яттю) відображається за адресою 0x1000 0000 і доступно лише за допомогою CPU через D-Bus.

Інтерфейс флеш-пам'яті здійснює доступ до флеш пам'яті через АНВ - шини I-Code і D-Code. Він реалізує стирання і програмні операції і механізми захисту читання / запису. Це прискорює виконання коду з системою попередньої вибірки і кешування лінії.

Флеш-пам'ять організована таким чином (рис.2.14):

- Основний блок пам'яті розділений на сектори
- Системна пам'ять, з якої пристрій завантажується в режимі завантаження пам'яті системи
- 512 ОТР (одноразово програмованих) байт для даних користувача.
- Опціональні байти для налаштування захисту читання і запису, сторожового таймера і скидання, коли пристрій знаходиться в режимі очікування або зупинки.

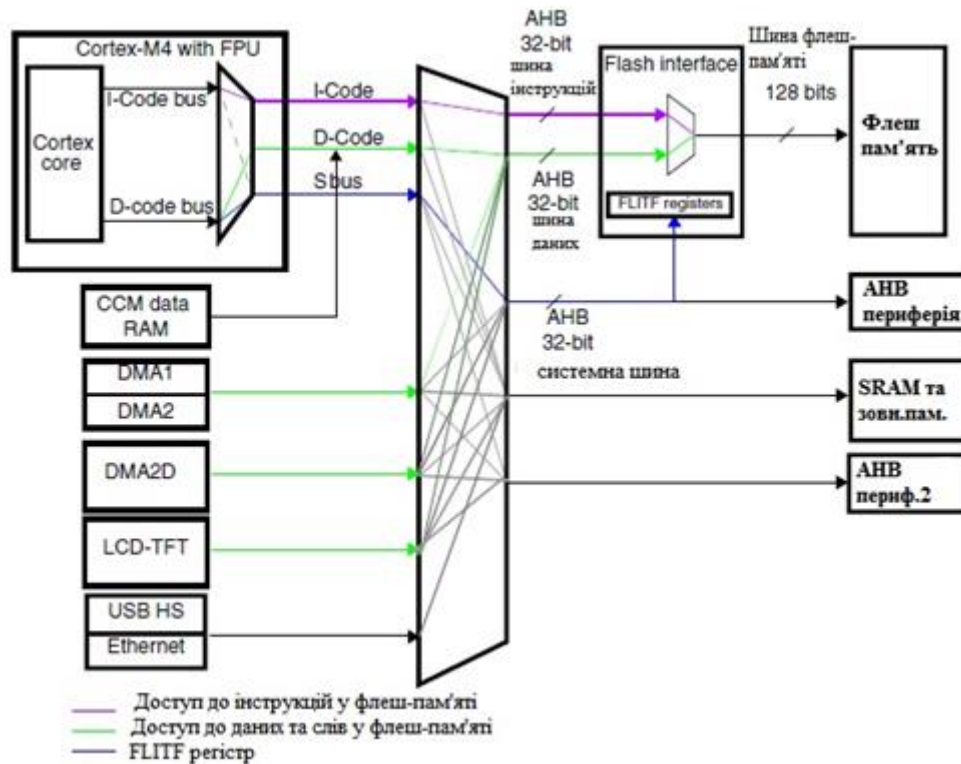


Рис. 2.14 Організація флеш-пам'ять

Конфігурація завантаження

У STM32F4xx, три різних режими завантаження можна вибрати через контакти BOOT [1: 0], табл. 2.2

Таблиця 2.2. Режими завантаження

Піни вибору режиму BOOT		Режими BOOT	Результат
BOOT1	BOOT0		
х	0	Головна флеш-пам'ять	Головна флеш-пам'ять обрана як завантаж. простір
0	1	Системна пам'ять	Системна пам'ять обрана як завантаж. простір
1	1	Вбудований SRAM	Вбудований SRAM обраний як завантаж. простір

Значення по контактам BOOT фіксуються на 4-му передньому фронті імпульсів SYSCLOCK після скидання. Залежить від користувача, як встановити прапорці boot1 і boot0 після скидання, щоб вибрати потрібний режим завантаження.

Boot0 є окремим піном, а boot1 сумісний з GPIO піном. Після того як boot1 був просемпльований, відповідний GPIO контакт є вільним і може бути використаний для інших цілей. Виходи BOOT також перезаписуються, коли

- HSI oscillator clock - внутрішній високочастотний генератор
- HSE oscillator clock - зовнішній високочастотний генератор
- PLL clock - ситема ФАПЧ

HSE oscillator clock - зовнішній високочастотний генератор

Джерелом сигналів для HSE генератора може бути як зовнішній тактовий генератор, так і звичайний кварцовий або керамічний резонатор.

Для встановленого на платі STM32-Discovery мікроконтролера частота зовнішнього сигналу не повинна перевищувати 24 МГц, при використанні зовнішнього тактового генератора, а при використанні кварцового (керамічного) резонатора його частота повинна бути від 4 до 24 МГц.

Зовнішній сигнал може мати форму пили, синусоїди або прямокутних імпульсів зі скважністю 50%.

HSI oscillator clock - внутрішній високочастотний генератор

HSI генератор являє собою RC-генератор з частотою 8 МГц, він тактує інтерфейс програмування флеш-пам'яті, може бути джерелом тактових сигналів (SYSCLOCK), а так само може служити джерелом опорних сигналів для ФАПЧ (однак при цьому його частота ділитися на два, т.е. становить 4 МГц).

Даний генератор проходить калібрування на заводі і виробник гарантує точність в 1% при температурі 25 градусів Цельсія.

PLL clock - ситема ФАПЧ

Система ФАПЧ (фазове автопідстроювання частоти) виробляє множення опорного сигналу (доступні коефіцієнти від 2 до 16), однак частота на виході системи ФАПЧ повинна лежати в межах 16-24 МГц.

Так само є два вторинних джерела тактових сигналів:

LSI RC - внутрішній низькочастотний RC-генератор (40 кГц)

LSE crystal - зовнішній низькочастотний кварцовий генератор (32 кГц)

Від внутрішнього низькочастотного RC-генератора тактується сторожовий таймер, так само від нього може тактуватися RTC-таймер, за сигналами якого можна виводити мікроконтролер із сплячого режиму. Всі ці джерела можна незалежно вмикати і вимикати, це особливо актуально коли необхідно знизити споживану потужність.

Тема 2.3. Порти введення/ виведення. Контролери NVIC та EXTI. Таймери

Порти введення/виведення

Безпосереднє управління станом пінів МК здійснюється за допомогою портів GPIO. В МК STM32 може бути до 11 незалежних 16-розрядних портів GPIO (General purpose input output), що позначаються літерами від A до K.

Порти можуть роботи в наступних режимах

- Input (вхід): Floating (плаваючий вхід з високим вхідним опором, нічим не навантажений), Pull-up (з верхнім навантажувальним резистором), Pull-down (з нижнім навантажувальним резистором)

- Output (вихід): Push-Pull двотактний ключ (Pull-up з верхньою навантаженням, Pull-down з нижнім навантаженням або по Pull просто двотактний вихід), Open Drain відкритий стік (Pull-up, Pull-down або по Pull). У режимі виходу може бути запрограмована швидкість роботи порту: 2 МГц, 25 МГц, 50 МГц або 100 МГц.
- Alternate Function (альтернативна функція): Push-Pull (Pull-up, Pull-down або по Pull), Open Drain (Pull-up, Pull-down або по Pull)
- Analog (аналогова лінія): цей режим необхідний, коли вивод використовується як канал ADC (АЦП) або вихід DAC (ЦАП).

Таким чином, при роботі на вхід можливі режими:

- Вхід – Hi-Z (Input floating)
- Вхід – підтяжка вгору (Input pull-up)
- Вхід – підтяжка вниз (Input-pull-down)
- Вхід – аналоговий (Analog)

При роботі порту на вихід :

- Вихід – з відкритим колектором (Output open-drain)
 - Вихід – двотактний (Output push-pull)
 - Альтернативні функції – вихід типу «з відкритим колектором» (Alternate function push-pull)
 - Альтернативні функції– двотактний вихід (Alternate function open-drain)
- Схемотехніка однієї лінії порту наведена на рис.2.16

or we.easyelectronics.ru

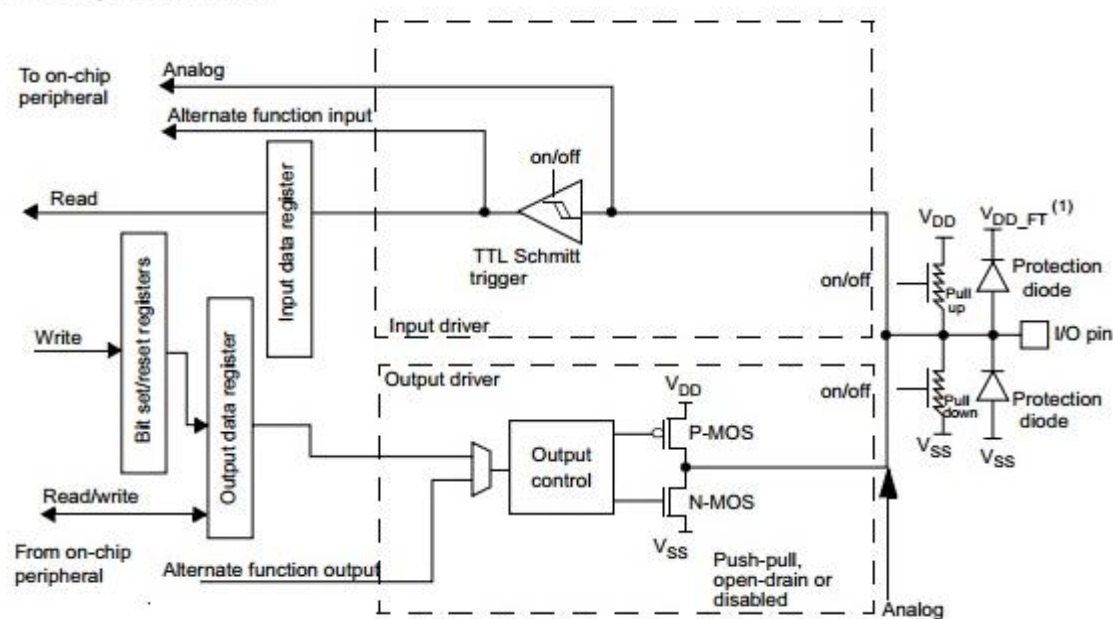


Рис.2.16 Структура одного розряду п'ятивольтового порту

V_{DD_FT} – потенціал п’ятивольтового порту (не збігається з V_{DD})

В режимі входу сигнал з піна I/Opin через триггер Шмітта поступає або на вхідний регістр (Input data register), або на вхід альтернативної функції (Alternate function input), або як аналоговий вхід (Analog). В режимі виходу дані з вихідного регістру (Output data register) або альтернативна функція (Alternate function output) поступає на вихідний каскад на компліментарній парі транзисторів, який може працювати в якості двотактного виходу або виходу з верхнім або нижнім навантаженням. На рис. 2.16 показано також керування верхнім та нижнім навантажувальними резисторами

Під час активного сигналу скидання і відразу після його закінчення всі альтернативні функції на виводах портів не активні (за винятком виводів, задіяних під JTAG), і всі порти виявляються зконфігурованими як плаваючі входи з високим опором.

Виводи генератора LSE (OSC32_IN і OSC32_OUT) можуть використовуватися як GPIO (PC14 і PC15 відповідно), якщо генератор LSE вимкнений. LSE має пріоритет перед функцією GPIO.

Виводи генератора HSE (OSC_IN і OSC_OUT) можуть використовуватися як GPIO (PH0 і PH1), коли генератор HSE вимкнений (основний кварцовий генератор не використовується, що буває дуже рідко). HSE має пріоритет над функцією GPIO.

Контролери NVIC та EXTI

Управління та обробка перериваннями виробляється контролером пріоритетних векторних переривань NVIC (Nested Vectored Interrupt Controller). Переривання та події (Interrupt and Events)

В МК Cortex-M є два поняття, які часто плутають - Interrupt і Event(переривання та подія).

Event - це подія (апаратна або програмна), на яку можуть реагувати ядро або периферійні блоки. Одним з варіантів реакції може бути - переривання.

Interrupt - це переривання роботи програми і перехід управління на спеціалізований ділянку - обробник переривання.

Кожне переривання викликається подією, але не кожна подія викликає переривання. Крім переривань, події можуть активувати і інші можливості МК, рис.2.17



Рис. 2.17 Переривання та подія

Джерела запитів переривань контролера NVIC наведено на рис.2.18

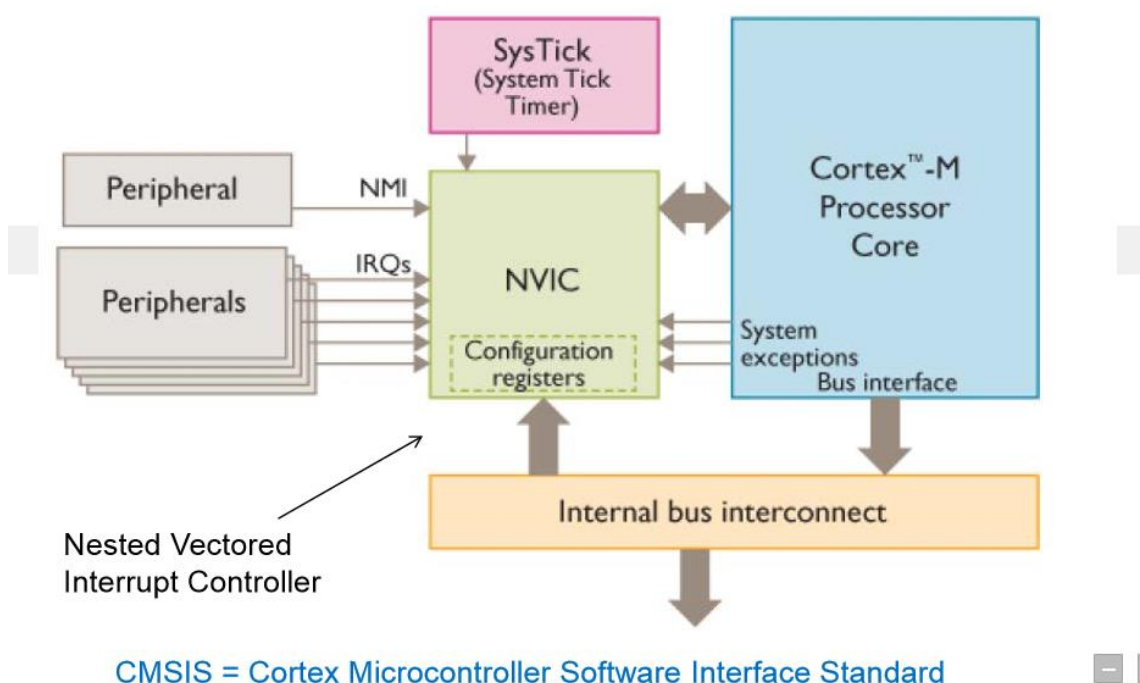


Рис.2.18 Джерела запитів переривань контролера NVIC

Як видно, на контролер NVIC приходять переривання від периферійних пристроїв МК, ядра і таймера. Зовнішні переривання, які поступають на піни МК, обробляються за допомогою використовується контролера зовнішніх переривань/ подій EXTI (External interrupt / event controller) Взаємодія його з NVIC ілюструє рис.2.19

23 edge detectors to trigger events and interrupts signaled by 240 GPIO pins and 7 internal events.

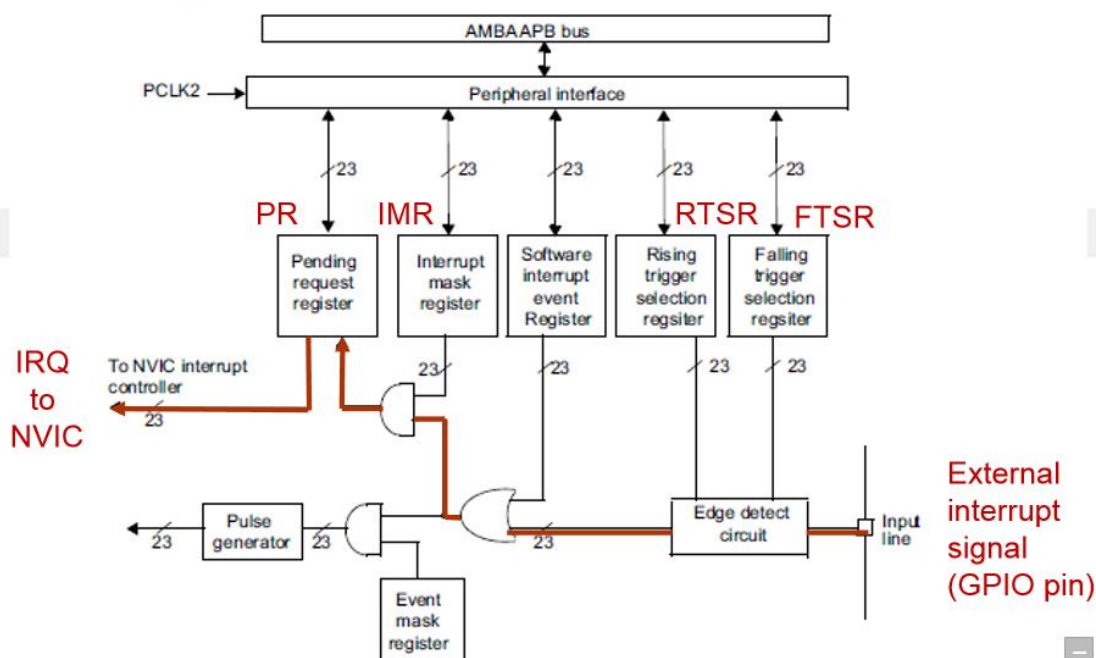


Рис.2.19 Взаємодія EXTI з NVIC

Зовнішнє переривання з піна порта GPIO pin через вибору детектування переднього або заднього фронту, схеми маскування та обробки режиму очікування поступає на вхід контролера пріоритетних векторних переривань NVIC.

Характеристика контролера NVIC наступна

- Загальна кількість переривань 240 (0-239)
- Програмований рівень пріоритету 0-255 для кожного переривання. Більш високий рівень відповідає нижчому пріоритету, тому рівень 0 має найвищий пріоритет переривання.
- Сигнали переривань можуть задаватися як рівнем, так і фронтами імпульсів.
- Динамічна зміна пріоритетів переривань.
- Угрупування пріоритетів по групам та підгрупам.
- Переривання у «хвості ланцюжка».
- Зовнішнє немасковане переривання Non Maskable Interrupt (NMI)
- Додатковий контролер WIC (Wakeup Interrupt Controller) забезпечує підтримку режиму сну з ультра-низьким енергоспоживанням.
- Процесор автоматично зберігає свій стан на вході переривань і витягує його зі стеку на виході без додаткових команд. Це забезпечує латентність обробки виключень.

Кожне переривання може знаходитися в одному з наступних станів:
Inactive (Неактивне): не активне і не прийняте.

Pending (В очікуванні): чекає обслуговування процесором.

Active (Активне): обслуговується процесором, але обслуговування не завершено
Active and pending (Активне і в очікуванні) обслуговується процесором, але в той же час очікується виключення від того ж джерела.

При виникненні деякої події контролер переривань автоматично перериває виконання основної програми, і викликає відповідну функцію обробки переривань. Після виходу з функції обробника переривань програма продовжує виконання з того місця, де відбулося переривання, рис. 2.20

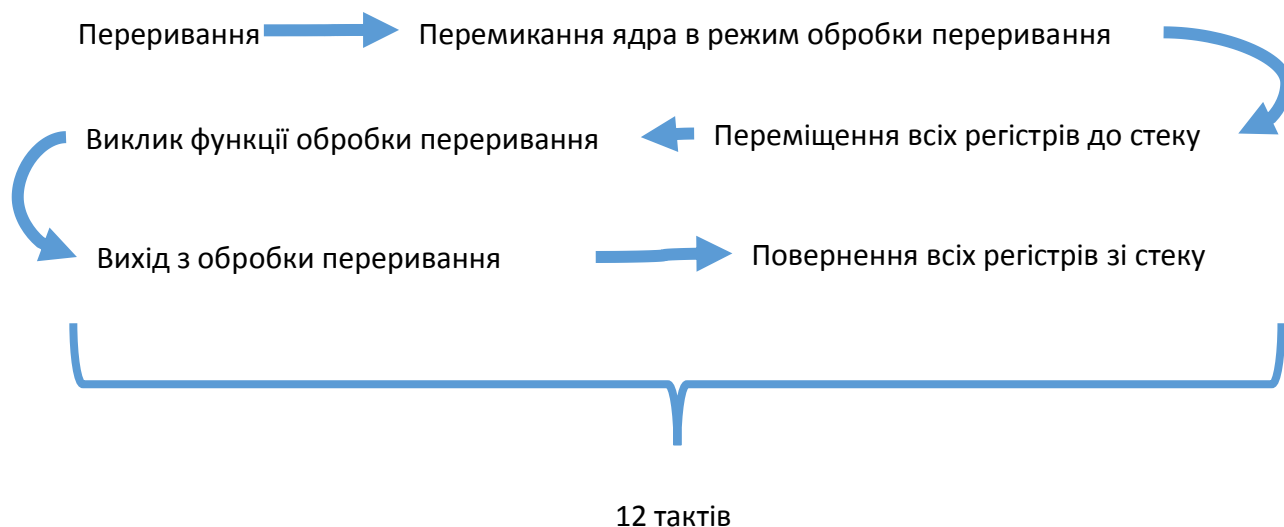


Рис. 2.20 Вхід в переривання і вихід з нього

Після перемикання ядра в режим обробки переривання регістри ядра поміщаються в стек. Безпосередньо під час запису значення регістрів в стек здійснюється вибірка початкової адреси функції обробки переривання. В стек переміщається наступні регістри - регістр статусу програми (Program Status Register (PSR)), лічильник програми (Program Counter (PC)) і регістр зв'язку (Link Register (LR)).(Опис регістрів ядра приведено в Cortex-M4 Generic User Guide та 2 лекції) Таким чином запам'ятовується стан, в якому перебувало ядро перед переходом в режим обробки переривань.

Також зберігаються регістри R0 - R3 і R12. Ці регістри використовуються в інструкціях для передачі параметрів, тому переміщення їх в стек робить можливим їх використання у функції обробки переривання, а R12 часто виступає в ролі робочого регістра програми.

По завершенні обробки переривання всі дії виконуються в зворотному порядку: витягується вміст стека і, паралельно з цим, здійснюється вибірка адреси повернення.

З моменту ініціації переривання до виконання першої команди обробника перериваних проходить 12 тактів, такий же час необхідно для поновлення основної програми після завершення обробки переривання (див. рис.2.20).

Контролер NVIC підтримує вкладеність переривань і пріоритети. Кожному перериванню при налаштуванні NVIC присвоюється свій пріоритет. Якщо під час обробки низько пріоритетного переривання виникає високо пріоритетне, то воно, в свою чергу, зупинить обробник низькопріоритетного переривання. NVIC підтримує переривання з різними пріоритетами, які можуть переривати один одного. При цьому, можуть виникнути різні ситуації, обробка яких по різному оптимізована:

1) Призупинення низькопріоритетного переривання

У цій ситуації, обробка низькопріоритетного переривання припиняється. Наступні 12 циклів виконується збереження в стек нового набору даних і запускається обробка високопріоритетного переривання. Після його обробки, вміст стека автоматично витягується і поновлюється обробка низькопріоритетного переривання.

2) Безперервна обробка переривань Ця ситуація може виникнути в двох випадках: якщо два переривання мають однаковий пріоритет і виникають одночасно або якщо низькопріоритетне переривання виникає під час обробки високопріоритетного.

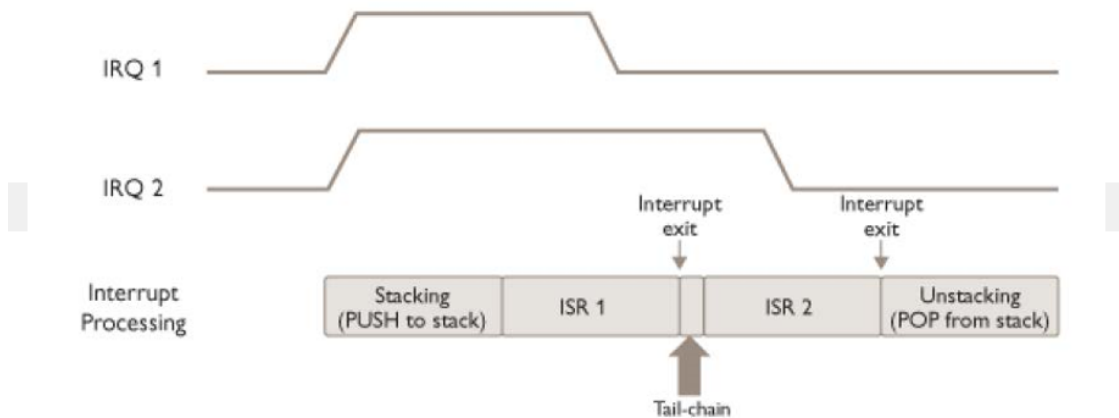
В цьому випадку, проміжні операції над стеком не здійснюються. Відбувається тільки завантаження адреси обробника наступного переривання і перехід до його виконання. Відмова від операцій над стеком економить 6 тактів, тобто перехід відбувається не за 12 тактів, а за 6.

3) Запізнення високопріоритетного переривання

Ситуація виникає, якщо високопріоритетне переривання відбувається під час переходу до низькопріоритетного. В цьому випадку перехід до високопріоритетного переривання відбуватиметься не менш ніж за 6 тактів з моменту його виникнення (час необхідний для завантаження адреси обробника переривання і переходу до нього).

Випадки 2 та 3 - переривання у «хвості ланцюжка»- пояснюється рис. 2.21

“Tail-chaining” interrupts



- NVIC does not unstack registers and then stack them again, if going directly to another ISR.
- NVIC can halt stacking (and remember its place) if a new IRQ is received.

Рис.2.21 Переривання у «хвості ланцюжка»

Два переривання IRQ1 і IRQ2 прийшли одночасно, IRQ1 внаслідок більш високого пріоритету отримала обслуговування. При поверненні із підпрограми IRQ1 і переході до підпрограми обробки IRQ2 стан основної програми не має сенсу зберігати в стеку і перехід відбувається безпосередньо.

Крім простої установки пріоритету переривань, NVIC реалізує можливість угруповання пріоритетів.

Переривання в групі з більш високим пріоритетом можуть переривати обробники переривань групи з нижчим пріоритетом. переривання з однієї групи, але з різним пріоритетом всередині групи не можуть переривати один одного. Пріоритет всередині групи визначає тільки порядок виклику обробника, коли були активізовані обидві події.

Для того, щоб вмикати / вимикати різні вектора переривань, існує маскування переривань. Маскування переривання здійснюється за допомогою регістрів Interrupt Set-enable Registers. Якщо переривання замасковано, це не означає, що периферія не генерує події! Просто NVIC не викликає обробник цієї події.

Таблиця векторів переривань містить адреси функцій обробників переривань і знаходиться в області пам'яті за 0 адресою. Номер у списку відповідає номеру переривання.

Таблиця 2.3. Вектори переривань

	Position	Priority	Type of priority	Acronym	Description	Address
STM32F4 Vector Table (partial) Tech. Ref. Table 61 (Refer to 'Startup Code')		-	-	-	Reserved	0x0000 0000
		-3	fixed	Reset	Reset	0x0000 0004
		6	settable	SysTick	System tick timer	0x0000 003C
	0	7	settable	WWDG	Window Watchdog interrupt	0x0000 0040
	1	8	settable	PVD	PVD through EXTI line detection interrupt	0x0000 0044
	2	9	settable	TAMP_STAMP	Tamper and TimeStamp interrupts through the EXTI line	0x0000 0048
	3	10	settable	RTC_WKUP	RTC Wakeup interrupt through the EXTI line	0x0000 004C
	4	11	settable	FLASH	Flash global interrupt	0x0000 0050
	5	12	settable	RCC	RCC global interrupt	0x0000 0054
	6	13	settable	EXTI0	EXTI Line0 interrupt	0x0000 0058
	7	14	settable	EXTI1	EXTI Line1 interrupt	0x0000 005C
	8	15	settable	EXTI2	EXTI Line2 interrupt	0x0000 0060
	9	16	settable	EXTI3	EXTI Line3 interrupt	0x0000 0064
	10	17	settable	EXTI4	EXTI Line4 interrupt	0x0000 0068
9	11	18	settable	DMA1_Stream0	DMA1 Stream0 global interrupt	0x0000 006C
	12	19	settable	DMA1_Stream1	DMA1 Stream1 global interrupt	0x0000 0070
	13	20	settable	DMA1_Stream2	DMA1 Stream2 global interrupt	0x0000 0074

Можна розмістити таблицю векторів переривань в іншій області пам'яті. За це зміщення таблиці векторів відповідає регістр таблиці векторів зсуву Vector Table Offset Register (VTOR).

Таймери

Мікроконтролер STM32 має у своєму складі кілька типів таймерів, що відрізняються один від одного за функціональним призначенням.

- базові (basic timers) (TIM6, TIM7)
- загального призначення (general-purpose timers) (TIM2-TIM5, TIM9-TIM14)
- просунуті (advanced-control timers) (TIM1, TIM8)

Крім того, в ядро процесора STM32 вбудований 24-бітний системний таймер, так званий SysTick.

Перший тип таймерів є найпростішим і являє собою базові таймери (Basic Timers). До даного типу належать таймери TIM6 і TIM7. Ці таймери дуже просто налаштовуються і управляються за допомогою мінімуму регістрів. Вони

здатні відраховувати інтервали часу і генерувати переривання при досягненні таймером заданого значення.

Другий тип являє собою таймери загального призначення (General-Purpose Timers). До нього відносяться таймери з TIM2 по TIM5 й таймери з TIM12 по TIM17. Вони можуть генерувати ШІМ, підраховувати імпульси на певних пінах мікроконтролера, обробляти сигнали від енкодера і т.п. Третій тип визначає таймери з розвиненим управлінням (Advanced-Control Timer). До цього типу належить таймер TIM1, який здатний виконувати всі перераховані вище операції. Крім того, на основі даного таймера можна побудувати пристрій, здатний управляти трифазним електроприводом. Таймери з розширеними функціями мають дуже широкий функціонал - комплементарні виводи для підтримки трифазних двигунів, підтримують режими рахунку в прямому і зворотному напрямках, генерацію ШІМ, канали захоплення / порівняння сигналу, режим одиночного імпульсу, підтримка DMA, додаткові функції безпеки у разі збоїв, підтримка інтерфейсу енкодера і датчика Холла. До шини APB1 підключені таймери TIM2, TIM3, TIM4, TIM5, TIM6, TIM7, TIM12; до шини APB2 - TIM1, TIM8, TIM9, TIM10, TIM11.

Базові таймери TIM6 та TIM7 побудовані на основі 16-бітових регістрів

Структурна схема базового таймера наведена на рис. 1. Попередній дільник TIMx_PSC дозволяє регулювати частоту тактових імпульсів для рахункового регістра, а регістр автозавантаження TIMx_ARR дає можливість задавати діапазон відліку таймера 2.22

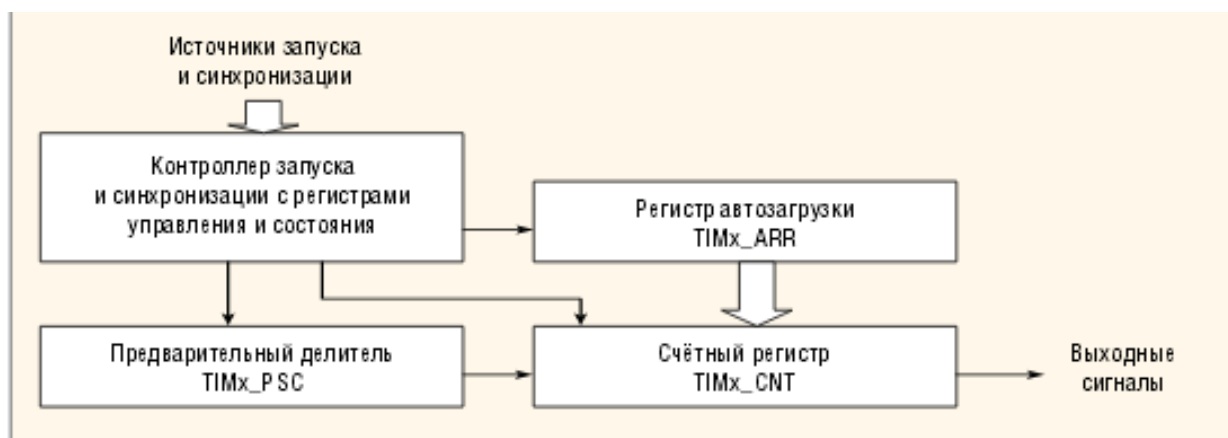


Рис. 2.22. Структурна схема базового таймера

Контролер запуску і синхронізації разом з регістрами управління і стану служить для організації режиму роботи таймера і дозволяє контролювати його функціонування.

Завдяки своїй організації лічильник таймера може рахувати в прямому і в зворотному напрямку, а також до середини заданого діапазону в прямому, а потім у зворотному напрямку. На вхід базового таймера може подаватися сигнал від декількох джерел, у тому числі тактовий сигнал синхронізації від

шини APB1, зовнішній сигнал або вихідний сигнал інших таймерів, що подається на висновки захоплення і порівняння. Таймери TIM6 і TIM7 тактуються від шини APB1. Якщо використовувати кварцовий резонатор з частотою 8 МГц і заводські настройки тактирования за замовчуванням, то тактова частота з шини синхронізації APB1 складе 24 МГц

Таймери загального призначення **General-Purpose Timers** мікроконтролера STM32F4xx мають більш широкі можливості, ніж базові, наприклад мають режими ШІМ, захоплення події, порівняння, тощо.

Тактування таймерів TIM2-TIM7, TIM12-TIM14 здійснюється генератором APB1; таймерів TIM1, TIM8-11 – APB2.

Тактова частота процесора SYSCLK визначається (за замовчуванням) частотою main PLL генератора (f_{SYSCLK} є основою АНВ1, АНВ2, АНВ3, APB1 та APB2 генераторів тактової частоти для периферії.) За допомогою дільників PLL_M, PLL_N та PLL_P встановлюється необхідне значення SYSCLK. Даний мікроконтролер може працювати на частоті до 180 МГц.

Просунуті таймери (advanced-control timers) (TIM1, TIM8)

Таймер TIM1 і провідні таймери загальнодоступних визначень TIMx мають повну незалежність і не мають ніяких загальних розподілених ресурсів; у кожного таймера є свій власний набір регістрів. З іншого боку, є механізми для здійснення взаємодії таймерів.

Основні можливості таймеру TIM1

- Реалізований на основі 16-бітового лічильника з автоматичною перезавантаженням початкового значення і з керованим напрямом рахунку (вгору, вниз або по черзі вгору / вниз).
- Має 16-бітовий програмований переддільник для тактового сигналу лічильника, що дозволяє здійснювати розподіл частоти на будь-який цілий коефіцієнт в межах 1..65536.
- Має до чотирьох незалежних каналів для введення / виведення сигналів з підтримкою кілька режимів роботи:
 - вхідна фіксація (Input Capture, тобто запис вмісту лічильника в регістр фіксації за сигналом на вході каналу; використовується для вимірювання періоду, тривалості імпульсів і інших тимчасових характеристик сигналу);
 - формування вихідного сигналу під управлінням схеми порівняння (Output Compare); логічний рівень сигналу на виході визначається в результаті зіставлення значень регістра порівняння і лічильника в даний момент;
 - генерація сигналу з широтно-імпульсною модуляцією (PWM, тобто ШІМ), з можливістю вибрати вирівнювання по краю або по центру імпульсу;
 - генерація вихідного сигналу у вигляді одиночного імпульсу.

- Має схему синхронізації для управління таймером зовнішніми сигналами і для забезпечення можливості внутрішнього взаємодії декількох таймерів мікроконтролера.

- Має лічильник циклів (repetition counter) для реалізації можливості поновлення регістрів таймера тільки після того, як відбудеться задана кількість циклів рахунку (переповнень лічильника).

- Має вхід зупинки (break input) для переведення вихідних сигналів каналів таймера (для каналів, що працюють в режимі виходу) в деякий зумовлене, безпечне для керованого зовнішнього обладнання стан (наприклад, при керуванні мостовим інвертором, це може бути стан, при якому всі ключі моста закриті).

- Забезпечує можливість генерації переривання або запиту DMA у відповідь на такі події:

- оновлення:

- при переповненні / антипереповненні лічильника, якщо при цьому лічильник циклів містить нульове значення;

- при програмної генерації події поновлення установкою біта UG в регістрі TIMx_EGR, якщо в регістрі TIMx_CR1 біти URS = 0, UDIS = 0;

- при генерації події поновлення по тригерній події, якщо в регістрі TIMx_CR1 біти URS = 0, UDIS = 0;

- тригерні (пускові) події - trigger event: запуск рахунку, зупинка, ініціалізації або рахунок за сигналом від внутрішнього або зовнішнього джерела;

- виконання вхідний фіксації (input capture) при роботі каналу в режимі входу;

- спрацьовування схеми порівняння (output compare) при роботі каналу в режимі виходу;

- сигнал на вході зупинки (break input).

- Має підтримку інкрементних датчиків положення (incremental encoder) і схемотехніки з датчиками Холла для вирішення завдань просторового позиціонування.

- Має вхід, який використовується в якості критичного (вхід для запуску заданого дії) або для поциклового управління струмом (cycle-by-cycle current management).

Таким чином, таймери вдосконаленого управління (TIM1, TIM8) можна розглядати як трифазні генератори ШІМ, мультиплексовані на 6 каналах. Вони мають додаткові ШІМ-виходи з програмованими часами пауз

Крім того, вони мають ті ж функції, що і загально- цільові таймери TIMx.

Їх 4 незалежні канали можна використовувати для:

- захоплення входу

- порівняння вихідних даних
- як 16-бітні генератори ШІМ (в різних режимах) з глибиною модуляції від 0-100%,
- як вихід з одним імпульсом

Таймер розширеного управління може працювати разом із таймерами TIMx за допомогою функції Timer Link для синхронізації або пов'язування подій. TIM1 і TIM8 підтримують незалежний DMA генерації запитів

Перелік посилань

1. Наикратчайшая всемирная история компьютеростроения с древних времен и до наших дней <http://infocom.uz/2004/01/19/naikratchayshaya-vsemirnaya-istoriya-kompyuterostroeniya-s-drevnih-vremen-i-do-nashih-dney/>
2. Презентація на тему ІСТОРІЯ ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ
<https://present5.com/prezentaciya-na-temu-istoriya-obchislyuvalno%D1%97-texniki-proekt-pidgotuvala/>
3. Э.П. Ланина История развития вычислительной техники. Иркутск, ИрГТУ, 2001 <http://84.237.19.2:8081/hoе/books/vt.pdf>
4. Комп'ютери п'ятого покоління
https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%27%D1%8E%D1%82%D0%B5%D1%80%D0%B8_%D0%BF%27%D1%8F%D1%82%D0%BE%D0%B3%D0%BE_%D0%BF%D0%BE%D0%BA%D0%BE%D0%BB%D1%96%D0%BD%D0%BD%D1%8F
5. Хронология развития вычислительной техники
https://ru.wikipedia.org/wiki/%D0%A5%D1%80%D0%BE%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%8F_%D1%80%D0%B0%D0%B7%D0%B2%D0%B8%D1%82%D0%B8%D1%8F_%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D0%B9_%D1%82%D0%B5%D1%85%D0%BD%D0%B8%D0%BA%D0%B8
6. Пятнадцать процессоров Intel x86, вошедших в историю
http://www.thg.ru/cpu/intel_cpu_history/index.html#386_32__
7. Історія розвитку мікропроцесорів
[https://uk.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BC%D1%96%D0%BA%D1%80%D0%BE%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D0%BE%D1%80%D1%96%D0%B2_I](https://uk.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BC%D1%96%D0%BA%D1%80%D0%BE%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D0%BE%D1%80%D1%96%D0%B2_I%80%D0%BE%D1%86%D0%B5%D1%81%D0%BE%D1%80%D1%96%D0%B2_I)
ntel
8. Комп'ютери п'ятого покоління
https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%27%D1%8E%D1%82%D0%B5%D1%80%D0%B8_%D0%BF%27%D1%8F%D1%82%D0%BE%D0%B3%D0%BE_%D0%BF%D0%BE%D0%BA%D0%BE%D0%BB%D1%96%D0%BD%D0%BD%D1%8F
9. Суперкомп'ютер
https://uk.wikipedia.org/wiki/%D0%A1%D1%83%D0%BF%D0%B5%D1%80%D0%BA%D0%BE%D0%BC%D0%BF%27%D1%8E%D1%82%D0%B5%D1%80%D0%BE%D1%86%D0%B5%D1%81%D0%BE%D1%80%D1%96%D0%B2_I
ntel
10. Мікропроцесорна техніка [Електронний ресурс] : підручник для студентів спеціальності «Електроніка» / В. Я. Жуйков, Т. О. Терещенко, Ю. С. Ямненко, А. В. Заграничний ; НТУУ «КПІ» ; ред. О. В. Борисов. –

- Електронні текстові дані (1 файл: 6,28 Мбайт). – Київ : НТУУ «КПІ», 2016. – 440 с <http://ela.kpi.ua/handle/123456789/18969>
11. Аляутдінов М. А., Галушкин А. В., Казанцев П. А., Остапенко Р. П. Нейрокомп'ютери: від програмної апаратної реалізації. — М.: Гаряча лінія — Телеком, 2008. — 152 с. — ISBN 978-5-9912-0044-8.
 12. ССС. Нейроемулятори
<https://www.victoria.lviv.ua/library/students/sss/theme2.html>
 13. NeuroMatrix® Core 3
https://module.ru/upload/files/nmc3_ryor_s_r_s_r_ryos_s_r_r_s_2012.pdf
 14. Американські фізики прогнозують нове покоління комп'ютерів
https://zik.ua/news/2019/07/18/amerykanski_fizyky_prognozuyut_nove_pokolinn_ua_kompyuteriv_16087152.
 15. Квантовий комп'ютер
https://uk.wikipedia.org/wiki/%D0%9A%D0%B2%D0%B0%D0%BD%D1%82%D0%BE%D0%B2%D0%B8%D0%B9_%D0%BA%D0%BE%D0%BC%D0%BF%27%D1%8E%D1%82%D0%B5%D1%80
 16. П. А. Белов, В. Г. Беспалов, В. Н. Васильев, С. А. Козлов, А. В. Павлов, К. Р. Симовский, Ю. А. Шполянский. Оптические процессоры: достижения и новые идеи. // В кн.: Проблемы когерентной и нелинейной оптики, СПб, 2006, стр. 6-36.
 17. Створений перший в світі діючий фотонний процесор
<https://cikavosti.com/stvoreniy-pershiy-v-sviti-diyuchiy-fotonniy-protseor/#hcq=0F1ufxr>
 18. ДНК-логіка как основа биокомпьютера
<https://compress.ru/article.aspx?id=17127>
 19. Оптичні нейронні мережі – основа надшвидких і надпотужних систем штучного інтелекту <https://tmginfo.net/2016/12/optichni-neyronni-merezhi-osnova-nadshv/>
 20. Cortex-M4 Generic User Guide
https://static.docs.arm.com/dui0553/a/DUI0553A_cortex_m4_dgug.pdf
 21. 1RM0090 Reference manual http://www.st.com/st-web-ui/static/active/en/resource/technical/document/reference_manual/DM00031020.pdf
 22. Cortex™ -M4 Devices Generic User Guide
http://infocenter.arm.com/help/topic/com.arm.doc.dui0553a/DUI0553A_cortex_m4_dgug.pdf
 23. RM0041 Reference manual
www.st.com/web/en/resource/technical/document/reference_manual/CD00246267.pdf
 24. AN4776 Application note General-purpose timer cookbook for STM32 microcontrollers

https://www.st.com/content/ccc/resource/technical/document/application_note/group0/91/01/84/3f/7c/67/41/3f/DM00236305/files/DM00236305.pdf/jcr:content/translations/en.DM00236305.pdf

25. AN5020 Application note Digital camera interface (DCMI) for STM32 MCUs

https://www.st.com/content/ccc/resource/technical/document/application_note/group0/c0/ef/15/38/d1/d6/49/88/DM00373474/files/DM00373474.pdf/jcr:content/translations/en.DM00373474.pdf

26. STM32F4Discovery – подключаем камеру по интерфейсу DCMI

<https://habr.com/ru/post/186980/>